ENABLING SMART CITY RESILIENCE: POST-DISASTER RESPONSE AND STRUCTURAL HEALTH MONITORING

BY

HYUNGCHUL YOON

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Civil Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Doctoral Committee:

Professor Billie. F. Spencer, Jr., Chair
Professor Gul Agha
Professor Mark E. Nelson
Professor Mani Golparvar-Fard

ABSTRACT

The concept of Smart Cities has been introduced to categorize a vast area of activities to enhance the quality of life of citizens. A central feature of these activities is the pervasive use of Information and Communication Technologies (ICT), helping cities to make better use of limited resources. Indeed, the ASCE Vision for Civil Engineering in 2025 (ASCE 2007) portends a future in which engineers will rely on and leverage real-time access to a living database, sensors, diagnostic tools, and other advanced technologies to ensure that informed decisions are made. However, these advances in technology take place against a backdrop of the deterioration of infrastructure, in addition to natural and human-made disasters. Moreover, recent events constantly remind us of the tremendous devastation that natural and human-made disasters can wreak on society. As such, emergency response procedures and resilience are among the crucial dimensions of any Smart City plan. The U.S. Department of Homeland Security (DHS) has recently launched plans to invest $50 million to develop cutting-edge emergency response technologies for Smart Cities. Furthermore, after significant disasters have taken place, it is imperative that emergency facilities and evacuation routes, including bridges and highways, be assessed for safety. The objective of this research is to provide a new framework that uses commercial off-the-shelf (COTS) devices such as smartphones, digital cameras, and unmanned aerial vehicles to enhance the functionality of Smart Cities, especially with respect to emergency response and civil infrastructure monitoring/assessment. To achieve this objective, this research focuses on post-disaster victim localization and assessment, first responder tracking and event localization, and vision-based structural monitoring/assessment, including the use of unmanned aerial vehicles (UAVs). This research constitutes a significant step toward the realization of Smart City Resilience.

ACKNOWLEDGEMENTS

First of all, I would like to deliver my sincere gratitude to and respect for my advisor, Professor Billie F. Spencer, Jr. for his guidance on my Ph.D. research as well as his strong support. Professor Spencer was not only an outstanding professor and mentor, but also a person who really cares about his students. I have learned from his academic insight, passion for conducting research, and teaching skills.

I would also like to express my thanks to my Ph.D. committee members, Professor Gul Agha, Professor Mark Nelson, and Professor Mani Golparvar, for providing me valuable advice and support toward the completion of my research.

When looking back on my Ph.D. journey, I owe a debt of gratitude to my colleagues with whom I've collaborated: Hazem Elanwar, Youngjib Ham, Soojin Cho, Reza Shefterfar, and Hajin Choi. I have enjoyed working with all of them, and they are part of the success of this research. I also want to acknowledge the assistance of and support from previous and current SSTL research group members, including Hongki Jo, Robin Kim, Sunghan Sim, Jian Li, Nick Wierschem, Lauren Lindeman, Brian Phillips, Fernando Moreu, Ryan Giles, Take Asai, Do-Soo Moon, Jongwoong Park, Gaston Fermandois, Yuguang Fu, Fernando Gomez, Brian Eick, Tu Hoang, Vedhus Hoskere, Yasutaka Narazaki, Jaeho Shin, Martha Cuenca, and Zach Treece. I've enjoyed academic and personal conversations with them during my journey through the completion of my master's and Ph.D. degrees.

Additionally, I want to thank to my friends at the University of Illinois, Donghyuk Jung, Paul Lee, Heng Chi, Jihoon Chung, Moochul Shin, Sung Woo Moon, Seung Jae Lee, Woobin Tark, Eugene Kim, SeungGu Kang, Byungmin Kim, Soon Pil Kang, Homin Song, Andy Hur, Seung Gu Kang, Roselyn Kim, and Daniel Lee who've studied together and shared their joy of life during

our respective time in Champaign. I would also thank Brian Y. Cho, Seungmin Lim, Junho Yang, Kyungmin Min, Kibum Noh, Seung Won Cho, and their families who attended the Korean Church of Champaign-Urbana.

The research in this dissertation has been supported by the National Science Foundation (Grants No. 1030454) and the American Association of Railroad (Richard Joy, program manager). I would like to gratefully acknowledge this support.

Finally, and most importantly, I would like to thank my family members, especially my wife Unmi Cha, for their boundless support and love. My two daughters Sarah and Sharon have given me so much joy and laughter. My parents, my sister and her husband (Jihye Yoon and Jaehoon Bang), and my brother-in-law Jaegyeong Cha have all provided strong support. My family has allowed me to overcome difficult times and complete my Ph.D. study.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

Our lives are invisibly interwoven with sensors connected through a continuous network that is often termed the Internet of Things. Evans (2011) forecast that there will be an average of seven connected devices for each person by 2020. Indeed, it is estimated that the number of connected devices have already surpassed the number of people on Earth (Riquier 2015). These smart sensors will enhance the life of human in many aspects, including shopping (Deloitte, 2011), transportation (Fowler 2015), industry (Manyika et al. 2011), and medicine (Howarth 2010).

Civil engineers are typically viewed as being quite traditional in their outlook, not taking full advantages of these advanced technologies. However, according to the American Society of Civil Engineers (ASCE) Vision for Civil Engineering in 2025 (ASCE 2007), civil engineers of the future will be relying on and leveraging real-time access to living databases, sensors, diagnostic tools, and other advanced technologies to ensure that informed decisions are made. These informed decisions are essential to guarantee the integrity of our infrastructure and quality of life, in the face of limited resources.

However, these advanced technologies are viewed against a backdrop of deterioration of our infrastructure and the occurrence of natural and human-made disasters. The recent ASCE Report Card (ASCE 2013) gave an overall GPA of D+ for infrastructure in the U.S., indicating that 24.9% of its bridges are structurally deficient and estimating that $3.6 trillion are needed to rehabilitate existing infrastructure. Moreover, recent events have shown the tremendous devastation that natural and human-made disasters can wreak on society. For example, the September 11 attack on the World Trade Center in the New Your City resulted in 2,996 deaths (Bram, Orr, and Rapaport 2002), the 2005 Hurricane Katrina caused in 1,836 deaths (Knabb,

Rhome, and Brown 2005), the 2008 Cyclone Nargis in Myanmar resulted in about 130,000 deaths (Webster 2008), the 2010 Haiti earthquake triggered about 316,000 deaths (Daniell, Khazai, and Wenzel 2013), and the 2011 Japan earthquake instigated a total of 15,861 deaths (Aoki et al. 2012).

The concept of Smart Cities has been introduced to categorize a vast area of activities to enhance the quality of life of a municipality's citizens. Seven key components of a Smart City were proposed by (Dirks, Keeling, and Dencik 2009): city services, its citizens, business, transport, communication, water, and energy. However, the resilience of civil infrastructure has not been emphasized. Resilience of Smart Cities requires that issues such as aging civil infrastructure and natural and human-made disasters need to be addressed as well.

The objective of this research is to provide new framework using commercial off-the-shelf (COTS) devices such as smartphones, digital cameras, and unmanned aerial vehicles for enhancing the functionality of Smart Cities, especially with respect to emergency response and civil infrastructure monitoring/assessment (Figure 1). To achieve this objective, this research focuses on post-disaster victim localization and assessment, first responder tracking and event localization, and vision-based structural monitoring/assessment, including the use of unmanned aerial vehicles (UAVs). The following paragraphs outline the key components of this research that are covered in this dissertation.

Chapter 2 provides background for this research. Previous research efforts on victim localization and assessment, first responder tracking and event localization, vision-based structural monitoring, and structural monitoring using UAVs are presented. Additionally, the limitations of this work are identified. An overview of Machine Learning methods, such as the Naïve Bayes Classifier is presented, focusing on pattern classification.

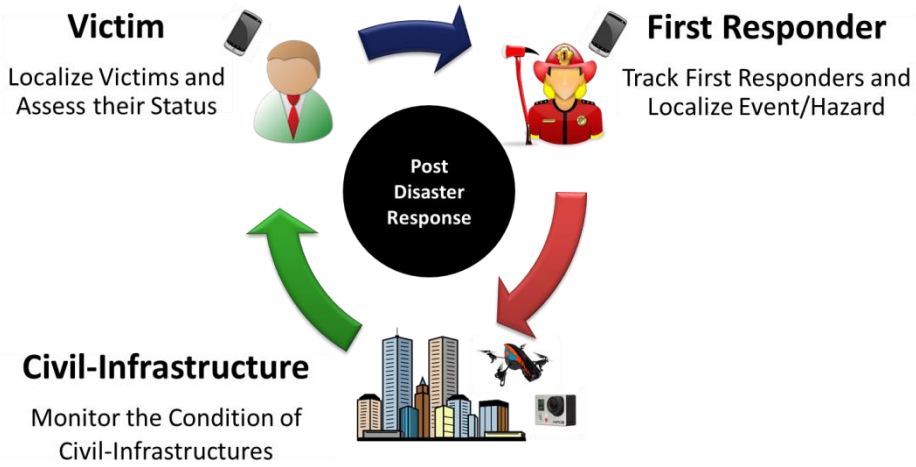


Figure 1: Objective of the proposed research.

Chapter 3 presents a new disaster response system to overcome the limitations of traditional search-and-rescue missions. Mobile devices such as smartphones can provide useful information for localizing victims and estimating their current status. A Victim Position System (VPS) localizes trapped victims at room level by using a Wi-Fi signal received from the smartphones, and a Victim Assessment System (VAS) estimates the status of victims by analyzing sensor data extracted from victims' smartphones. The accuracy of both the VPS and the VAS systems is validated using an in-building test assuming a disaster scenario.

In Chapter 4, a new algorithm to track first responders, and to localize important events such as fire or structural hazards more accurately is presented. Because the WLAN-based localization method only works for room level accuracy, a dead-reckoning-based tracking method is proposed for finding the location of first responders. Furthermore, to reduce accumulation error in the dead-reckoning method, a Forward-Backward approach is proposed to accurately localize an important event for rescue missions. The efficiency and accuracy of the proposed system is validated by the use of an in-building test assuming a disaster scenario.

Chapter 5 presents a new algorithm for system identification using commercial cameras, such as the camera embedded in smartphones. Computer-vision-based methods, including detecting natural features from a structure and tracking features, are presented. This chapter investigates the potential of using commercial cameras for structural system identification, and is validated by analysis in a laboratory setting of a six story structure.

Chapter 6 presents a new approach that utilizes unmanned aerial vehicles (UAVs) to monitor civil infrastructures. While Chapter 5 assumes that a camera is stationary, the cameras installed to UAVs are non-stationary. This chapter investigates using the UAVs for measuring the displacement of civil infrastructures by proposing a method for removing the non-stationary motion of a camera from video taken by a UAV.

Chapter 7 summarizes the research presented in this dissertation, and discusses future studies for the purpose of extending the proposed work into other engineering domains.

CHAPTER 2 BACKGROUND

This chapter describes previous literature related to the proposed research. Topics covered include: victim localization and assessment, first responder tracking and hazard localization, and structural health monitoring using unmanned aerial vehicles (UAVs). The literature review begins with general indoor localization techniques, and how they can be used for emergency response. Then, activity recognition using sensors will be discussed and how these methods can be extended to assess the status of victims. Next, dead-reckoning, a process for calculating one's position by using a previously determined position, will be discussed, as well as how these methods can be used to track first responders and locate structural hazards. The current state of the art of structural health monitoring will be then discussed. Finally, efforts to identify the condition of structures using UAVs will be introduced. The goal of this chapter is to identify gaps in knowledge that must be filled to develop more efficient post disaster response and recovery systems.

2.1 Victim Localization and Assessment

Current search-and-rescue efforts are conducted by employing three main methods: physical search, canine search, and electronic search (FEMA 2014). Through the physical void search that is used in most incidents, first responders use visual and vocal assessment to locate victims. Given that this method does not require any specialized electronic equipment, it will invariably overlook unconscious victims. Therefore, this method is restricted to areas in which first responders can safely enter and is highly affected by site conditions, such as visibility, temperature, time of the day, etc. To locate unconscious victims, a well-trained canine team can be effective. However, canine search is still affected by accessibility to the site and the overall capability of the canine

team. Furthermore, canines must take a rest periodically (e.g., 20~30 minutes of search followed by 20~30 minutes of rest). In some cases, electronic listening devices are deployed in search-and-rescue efforts. These devices are able to cover large areas, detecting acoustic noises and vibrations coming from the victims. The electronic search also has limitations in detecting unconscious victims and difficulty in deployment and monitoring on site.

Due to the limitations of existing search and rescue systems, considerable demand has emerged for a system that provides first responders with accurate and immediate information at a disaster site. The author has visited the Illinois Fire Service Institute to understand the most important information necessary for supporting first responders. The most immediate demand was for a device to localize and quickly assess victims' status, regardless of their state of consciousness. As described in the next section, there has been much effort to develop the systems to locate victims and assess their physical status inside buildings.

Recently, a few studies have been reported using advanced technologies for emergency response operations. Peña-Mora et al. (2010) developed an information/technology-based collaboration framework that supports civil engineering emergency response operations. Within this framework, the critical building information is sent to first responders through digital devices via a wireless and ad-hoc network. Rantakokko et al. (2011) proposed a concept for an indoor localization system for first responders using multiple sensors, such as GPS, magnetometers, barometers, imaging sensors, ultrasonic sensors, etc. In their paper, possible technologies for indoor localization were surveyed. Li et al. (2014) proposed an environment aware beacon deploy algorithm to enhance a sequence-based localization of trapped victims in fire situations. With integration of building information modeling (BIM) and metaheuristics, the beacons are optimally distributed to locate people in the building under fire. While this research provided significant

directions for improving current emergency response systems, it has have not provided a prototype that is ready to be deployed in the field.

Regarding the indoor localization system, the first trial was a radio frequency based indoor user localization system, named RADAR (Bahl and Padmanabhan 2000). Many researchers have pursued similar approaches for indoor localization (Hatami 2006; Zàruba et al. 2007). Researchers have focused on the use of Wi-Fi signals, as Wi-Fi is now available in most indoor environments (Van Haute et al. 2013). Many researchers have employed the Wi-Fi-based localization to smartphones, given the increasing us of such phones, which have been demonstrated for successful localization performance (Kothari et al. 2012; Link et al. 2011; Liu et al. 2012; Martin et al. 2010; Pei et al. 2012; Subbu, Gozick, and Dantu 2013; Yim 2013). With regard to physical status assessment, there have been many studies that recognize activities that can be used to interpret physical status. This recognition has been carried out, based on measurements using embedded sensors, and the mostly commonly used measurement has been acceleration (Ravi et al. 2005). Activity recognition techniques have been extended to utilize various embedded sensors in smartphones, as carried out by many researchers (Khan et al. 2010; Keally et al. 2011; Lee and Cho 2011; Weiss and Lockhart 2012; bin Abdullah et al. 2012; Yan et al. 2012). Though the systems showed good performance in localization and assessment of people situated inside buildings, these developments have not been tailored for disaster situations, such as localization under some Wireless Access Points that have been damaged or recognition of activities used to assess victims' status during a disaster.

Pattern classification algorithms have been widely employed in both activity recognition and indoor localization systems (Ravi et al. 2005; Parnandi et al. 2010; Khan et al. 2010; Lee and Cho 2011; bin Abdullah et al. 2012; Pei et al. 2012). Among various algorithms, the K-nearest-

neighbor (KNN) classifier and the Naïve Bayes classifier are the most widely used techniques due to their simplicity and effectiveness. The KNN classifier calculates the Euclidean distance between the test input and the labeled training samples, and the test input is classified into the most frequent class among *k* nearest training samples (Cover and Hart 1967). Though the KNN classifier is relatively simple, it can be computationally expensive for large training sets, mostly due to the need for calculating all distances between the test input and the entire training set. This time-consuming process is not appropriate for near real-time estimation using streamed inputs. On the other hand, the Naïve Bayes classifier is a simple probabilistic classifier based on Bayes' theorem, with a naïve independence assumption—namely, that each feature contributes independently to the probability that an input belongs to a class (Duda, Hart, and Stork 1999). The Naïve Bayes classifier is based on assumption that all features are conditionally independent; however, the classifier can also be used when the features have some dependencies (Rish 2001). The Naïve Bayes classifier is found to be very efficient in many complex real-world situations (Zhang 2004); the parameters for the Naïve Bayes classifier can be obtained by calculating the mean and variance of the training data. Once the required parameters are obtained, they are stored and later used to estimate the testing data with minimal computation. The Naïve Bayes classifier is efficient in terms of data storage and computing resources, while also displaying a high rate of success. Therefore, the Naïve Bayes classifier is employed in Chapter 3 for both localizing and assessing the status of the victims.

Consider the task of determining the status of a victim given measured data from smartphone sensors. In the language of pattern classification, the victim status is termed the "class" and is designated as *C*; "features", denoted *F*, are specific characteristics extracted from the measured data. Assume that there are *m* classes (i.e., victim states) and *n* features (feature selection

will be discussed further in later sections). The Naïve Bayes classifier uses Bayes' theorem to determine the most likely class (victim status), given features. The probability of a class *C* being c_i (i.e., the likelihood that the state of a victim is c_i) can be written as,

$$\mathrm{P}\big(C = c_i \big| F_1, \dots F_j, \dots, F_n\big) = \frac{\mathrm{P}\big(F_1, \dots F_j, \dots, F_n \big| C = c_i\big)\mathrm{P}(C = c_i)}{\mathrm{P}(F_1, \dots F_j, \dots, F_n)} \tag{1}$$

where F_j is the j^{th} feature variable. Using conditional probability, the numerator of Eq. (1) can be rewritten as,

$$\begin{aligned} &\mathrm{P}(C = c_i)\ \mathrm{P}\big(F_1, \dots F_j, \dots, F_n \big| C = c_i\big) \\ &= \mathrm{P}(C = c_i)\ \mathrm{P}(F_1 | C = c_i)\mathrm{P}\big(F_2, \dots F_j, \dots, F_n \big| C = c_i\big) \end{aligned} \tag{2}$$

The Naïve Bayes rule follows the assumption that all features are conditionally independent to every other feature. Then the Eq. (2) can be written as,

$$\begin{aligned} &\mathrm{P}\big(C = c_i \big| F_1, \dots F_j, \dots, F_n\big) \\ &\propto \mathrm{P}(C = c_i, F_1, \dots, F_n) \\ &\propto \mathrm{P}(C = c_i)\mathrm{P}(F_1 | C = c_i)\mathrm{P}(F_2 | C = c_i) \dots \mathrm{P}(F_n | C = c_i) \\ &\propto \mathrm{P}(C = c_i) \prod_{j=1}^{n} \mathrm{P}\big(F_j \big| C = c_i\big) \end{aligned} \tag{3}$$

$$\mathrm{P}(C = c_i|F_1, \dots, F_n) = \frac{1}{\mathrm{Z}}\mathrm{P}(C = c_i)\prod_{j=1}^{n}\mathrm{P}\left(F_j\middle|C = c_i\right) \quad (4)$$

where Z is the denominator of Eq. (1), which can be neglected during the classification because the probability of the features can be assumed to be constant.

$\mathrm{P}\left(F_j\middle|C = c_i\right)$ where j = 1… n can be obtained by assuming an underlying Gaussian probability distribution and using the training data to estimate the associated parameters. From the Bayes' rule, the estimated class will be the one that has the highest value in the product of the conditional probability of every feature, i.e.,

$$c^* = arg\max_{c_i}\mathrm{P}(C = c_i)\prod_{j=1}^{n}\mathrm{P}\left(F_j = f\middle|C = c_i\right) \quad (5)$$

where c_i is the classified class and f is the feature value of the test data.

The Naïve Bayes classifier outlined in this section will be used for both the VPS and VAS modules to classify the location and the status of the victim. First, the training data will be used to determine the parameters for the Gaussian probability $\mathrm{P}\left(F_j\middle|C = c_i\right)$; subsequently, using the Bayes' Theory and by assuming statistical independence of all features, $\mathrm{P}(C = c_i|F_1, \dots, F_n)$ will be calculated using Eq. (4). Finally, the state will be classified as the one that has the maximum probability shown as Eq. (5). This approach will provide the expected location and status of the victim.

2.2 First Responders Tracking and Event/Hazard Localization

Because the WLAN based indoor localization methods are only valid for room-level accuracy, another method is needed to track first responders in open spaces, such as hallways. Dead-reckoning is one of the solutions for tracking first responders, as it determines a user's current location based on previously determined locations and advances the position of the user based upon previously known or an estimated speed established for a given time period (Ojeda and Borenstein 2007; Beauregard and Haas 2006). By leveraging a wide range of sensors, several studies have looked into improving the reliability of localization using the dead-reckoning process. For instance, Akula et al. (2011) proposed a location tracking method that integrates GPS with a dead reckoning mechanism. By leveraging pre-determined locations of a user and manual user interventions, their method minimizes the drift errors in motion trajectory—errors that typically grow in proportion to the moving distance of the users. Despite promising results, their method (similar to (Chengliang, Zaiyi, and Lian 2010) requires an external Inertial Measurement Unit (IMU) sensor attached to a user's shoe. Similarly, in the case of (Beauregard and Haas 2006) and (Collin, Mezentsev, and Lachapelle 2003), the IMU was mounted on a helmet or carried in a backpack respectively.

Thanks to growing demands of the consumer market, sensors—including accelerometer, gyroscope, magnetic field, pressure and various other types—are today available in the majority of commodity smartphones and can assist with the task of user localization. Dead reckoning using a smartphone has several advantages. For example, it does not require external sensors, which is beneficial from a mobility perspective. It also does not adversely impact a user's sense of comfort within unobtrusive sensing environments. Several studies (Pratama, Widyawan, and Hidayat 2012; Pai et al. 2012; Han et al. 2012; Tian et al. 2014) report promising results upon deriving user's

location information solely based on sensory data from a commodity smartphone, such as an accelerometer or an electronic compass. Nevertheless, anecdotal evidence and controlled studies (Beauregard and Haas 2006; Ojeda and Borenstein 2007) conducted on dead reckoning-based positioning systems have raised concerns about the accumulation of the drift errors in proportion to the traveling distance of the users.

2.3 Post-Disaster Structural Condition Assessment and Structural Health Monitoring

In addition to the location and the status of the victim, identifying structural hazards and monitoring the condition of civil infrastructures are also open tasks for civil engineers in post-disaster response and recovery (Peña-Mora et al. 2008). Federal agencies in the U.S. have provided some guidelines for structure assessment —The U.S. Federal Emergency Management Agency (FEMA) developed a rescue field operation guide to provide a national standard rescue system (FEMA 2006). The guide is designed to supplement the National Urban Search and Rescue Response (US&R) System, and includes a method for marking the condition of structures (Figure 2). However, these assessments are being done by visual means of inspection and therefore could be heavily subjective. While the manual by FEMA is meant to be used for rescue processes, the Applied Technology Council (ATC) has developed manuals designed to create a better recovery process after both natural and human-made disasters such as earthquakes, windstorms, hurricanes, snow storms, fires, floods, tsunamis, or terrorist attacks (ATC 2003). While the rapid and detailed evaluation processes mainly depend on visual information, the engineering evaluation requires a detailed investigation of damaged structures including construction drawings and damage plans (Figure 3). However, this detailed investigation procedure requires resources and large amounts of time.

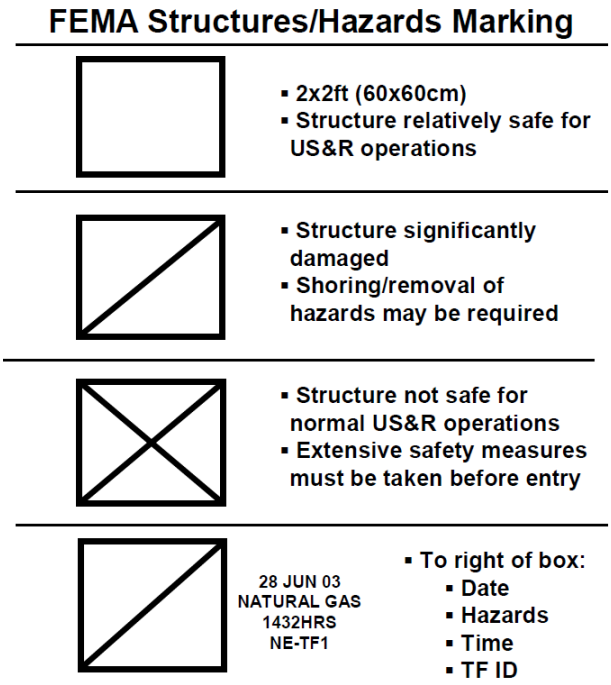
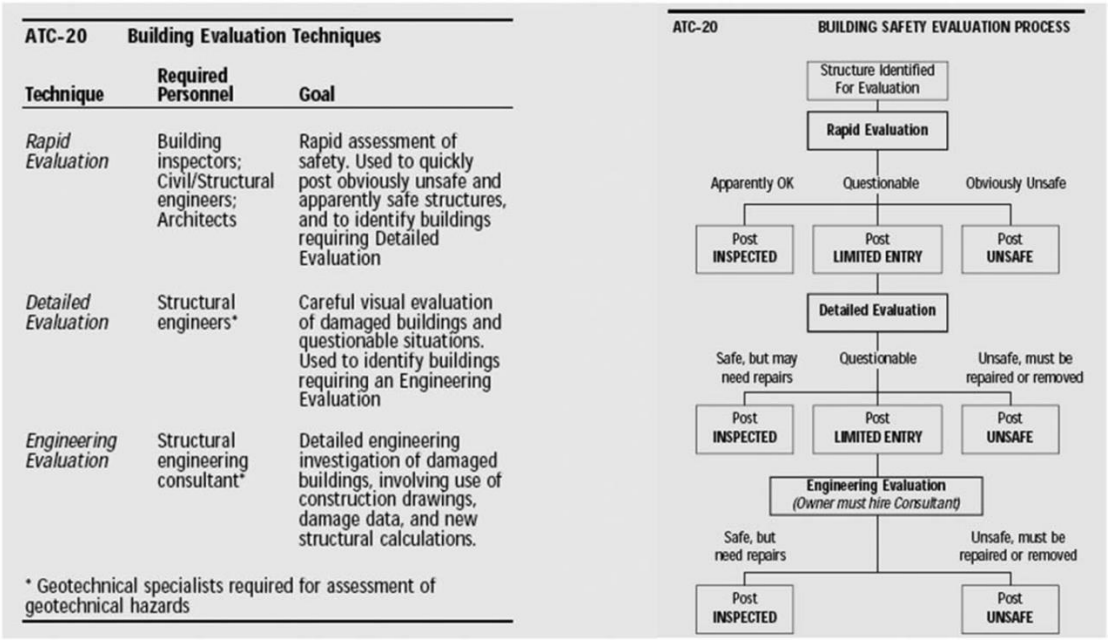


Figure 2: FEMA Structure/Hazards Marking (FEMA 2006).



ATC-20 Building Evaluation Techniques

Technique	Required Personnel	Goal
Rapid Evaluation	Building inspectors; Civil/Structural engineers; Architects	Rapid assessment of safety. Used to quickly post obviously unsafe and apparently safe structures, and to identify buildings requiring Detailed Evaluation
Detailed Evaluation	Structural engineers*	Careful visual evaluation of damaged buildings and questionable situations. Used to identify buildings requiring an Engineering Evaluation
Engineering Evaluation	Structural engineering consultant*	Detailed engineering investigation of damaged buildings, involving use of construction drawings, damage data, and new structural calculations.

* Geotechnical specialists required for assessment of geotechnical hazards

(a)

(b)

Figure 3: (a) building evaluation techniques and (b) building safety evaluation process (ATC 2003).

Bridges, in particular, are considered critical infrastructure with respect to the number of casualties caused by disasters. One hundred and fifty-one people were killed due to the collapse of the Tangiwai bridge in New Zealand in 1953(Conly and Stewart 1986). Forty-two people were killed resulting from the collapse of the Cypress Street Viaduct in Oakland, CA (Yashinsky 1998), and 20 people were killed during the collapse of the Kutai Kartengara Bridge in Indonesia (Chen and Omenzetter 2013). Due to the importance of bridges, FEMA (2004) has announced a checklist for damage assessment, including bridge and damage descriptions. Bridge descriptions such as length, type, number of lanes, and width can be obtained by drawings, but could be changed due to modifications or disasters. On the other hand, damage of a bridge can be obtained by visual inspection and structural health monitoring (SHM) systems. However, most bridges do not have SHM systems, and a high level of risk exists when performing detailed bridge inspections and installing sensors after a disaster occurs. Therefore, a safer and more efficient method for bridge assessment is needed.

Structural health monitoring primarily contributes sensing and measurement, which comprise a critical topic in various engineering fields. The research in the field of civil engineering also requires data acquisition of structural behaviors such as displacements, accelerations, and strains. Displacements, for example, are usually measured by linear variable differential transformer (LVDT) if structures have relatively large deformation. In cases of accelerations and strains, accelerometers and strain gauges accurately measure the dynamic motion of structures by means of the piezoelectric effect. Although these sensors have enough sensitivity to capture the structural behavior, the experimental step requires an installation progress directly attached at the measurement point and it is not appropriate for almost large-scale civil engineering structures in the field (Fukuda et al. 2013). Furthermore, additional data acquisition devices and accompanying

long wires make this measurement procedure tedious. To overcome the limitation of the wired sensors, wireless sensor network has been utilized to measure the dynamic response of structures (Spencer, Ruiz-Sandoval, and Kurata 2004). While the enhancements in the hardware and software aspects have improved the performance of wireless sensors, these methods are still challenged by the need for manual installation of the sensors.

Recent computer vision-based techniques provide an opportunity to measure the dynamic movement of the structures, with minimal effort. Compared to conventional measurement instruments, these methods do not require installation and maintenance of expensive sensor setups. Some of their early examples of these methods were provided in the work of Nogueira, Barbosa, and Barra (2005); Chang and Xiao (2010), Khalil (2011), and Lee, Ho, and Lee (2012) all of whom presented a method for displacement measurement using specially designed targets. Morlier (2011); Ji and Chang (2008); and Caetano, Silva, and Bateira (2011) employed the Optical Flow-based method to automatically measure structural displacement from a sequence of images. Shih and Sung (2013); and Kim et al. (2013) leveraged the digital image correlation (DIC) technique (Hild and Roux 2006) to extract displacements of structures and show an example of characterizing the vibration mode of a cantilever beam. Correlated Solutions, Inc., introduced a commercialized package, the VIC-2D system that utilizes digital image correlation algorithms to measure in-plane displacement. Schumacher and Shariati (2013) used virtual visual sensors and measure the rate of change in the intensity of certain points to characterize dynamic motions for simple structures, in the laboratory and field conditions. Fukuda et al. (2013); Feng et al. (2015) used template matching technique called OCM to track points without a target panel.

Despite success in preliminary experiments, most of these vision-based measurement techniques still have one or more limitations. First, these techniques either require installation of

targets, which makes the process tedious, or employ a template matching algorithm, which may be computationally costly. Second, most of these techniques requires relatively high-speed and high-resolution cameras, or requires additional acquisition equipment and lenses, which may be prohibitively expensive. Moreover, issues such as temporal aliasing and sampling frequency variations have not been discussed. Finally, most vision-based methods have been developed based on the use of a stationary camera—that is, assuming that cameras should in fixed position. However, for instance, cameras are subjected to multiple degrees of freedom motion when attached to unmanned aerial vehicles.

2.4 Summary

This chapter has provided the necessary background for exploring general needs for post-disaster response and structural monitoring using advanced technologies. Additionally, it has identified gaps in the research in this field. Indoor localization and activity recognition methods were introduced, but were not tailored for post-disaster response and recovery. For indoor localization, the WLAN fingerprinting-based technique can be adopted for localizing trapped victims inside a building. However, the current technique is only valid for a normal condition, which is not applicable for a post-disaster situation. Activity recognition can be adopted to estimate the status of the victim. To track the first responders and locate some important events such as structural hazards, fires, or shooters inside a building, the dead-reckoning technique can be used. Much research has been done on the subject of dead-reckoning, but researchers were not able to solve the drift error which can be accumulated by distance. Finally, for structural monitoring, conventional meth SHM systems and techniques were reviewed. Infrastructures that already have wired or wireless SHM systems installed could be monitored easily; however, most infrastructures will not have such systems installed. Installation of

sensors is not a trivial process, thus conventionally wired and wireless SHM systems are not appropriate. Recently, computer vision-based SHM techniques were introduced, but most of the work still requires additional targets to be installed. In addition, these techniques assumed to take video from fixed ground, which will bring inaccurate results when used together with UAVs.

CHAPTER 3 VICTIM LOCALIZATION AND ASSESSMENT

In this chapter, a smartphone-based, in-building victim localization and assessment system is developed. Because a GPS cannot be utilized in an indoor environment, a WLAN-based indoor localization system will be adopted to localize a victim trapped inside a building. WLAN-based localization uses the received signal strength indicator (RSSI) of Wi-Fi signals from multiple wireless access points (WAP), while referencing a pre-established Wi-Fi fingerprinting map of a building. The localization technique is improved in this study by implementing a Naïve Bayes classifier for localization in between the statistically measured points and in the context of a disaster situation in which some of the WAPs might not work properly.

The victim assessment system is designed to estimate the status of a victim using 3D The victim assessment system is designed to estimate the status of a victim using 3D acceleration measurements from a smartphone. Six features that can be distinguished from the measured acceleration profiles with minimal computation are selected and linked with basic activities (e.g., sitting, lying, walking, running, etc.) of a victim. A Naïve Bayes classifier is again employed to compare the features from the measured acceleration, with training data stored on the smartphone. A victim's status (i.e., highly ambulatory, ambulatory, non-ambulatory, and unconscious) is inferred by first responders with information taken from continuously calculated activities.

Location and status information may be transmitted to first responders by using portable WLAN (Wireless Local Area Network) temporarily built at the disaster site. This timely information, obtained by the proposed system and sent to first responders, is expected to significantly improve the rescue process with regard to accuracy, reliability, and safety.

A WLAN-based indoor localization can localize the victims trapped in the building with room-level accuracy, but is not suitable for tracking first responders, or locating important events such as structural hazards, fires, and shooters in a building. The accuracy of the WLAN-based localization is heavily dependent on signal barriers, such as walls. In open spaces such as corridors, the WLAN-based system will not be able to locate the user. Dead-reckoning, as described in section 2, is one of the solutions for tracking first responders. The process can track the user with simple equipment such as sensors embedded in smartphones, but has the limitation of a drift error that is proportional to the travel distance.

To address the current limitations of dead-reckoning, this task presents a new infrastructure-free approach for 3D event localization in indoor and outdoor GPS-denied environments. Here, event localization refers to "finding the location of a user when conducting any value-added activities in emergency response," such as locations of the structural hazard, fire, and shooter in a building. To do so, the proposed method leverages existing sensors that are embedded in a commodity smartphone and does not require the user to carry additional hardware. Using data obtained from these sensors, multiple 3D dead-reckoning paths are integrated into a probabilistic model. I hypothesize that the probabilistic integration of the localization results from multiple dead-reckoning processes based on different reference points will produce more reliable localization results compared to single best estimations from a one-way dead-reckoning process. In the following section, the proposed probabilistic approach for 3D event localization and the underlying algorithms using embedded sensors in a smartphone are presented. Then, experimental results based on several case studies on a multistory parking garage are presented , all of which validate our hypothesis. Perceived benefits and open research challenges are also discussed in detail.

3.1 Victim Localization

Development of the Victim Positioning System (VPS)

The VPS relies on Wi-Fi signals from Wireless Access Points (WAPs) to find the locations of smartphones inside a building following a disaster. Many buildings, both residential and commercial, have several WAPs that can be leveraged for the tracking of indoor location. The VPS is developed in addition to the assumption that everyone in the building has a Wi-Fi-enabled smartphone, and the number of detectable WAPs is three or more anywhere inside the building. It is further assumed that many of these WAPs survive minor to moderate disasters and continue working.

The VPS is developed upon a well-known WLAN-based indoor localization system called RADAR. The RADAR uses the presence of the WAPs in addition to the received signal strength indicators (RSSI) of the detected WAPs in the building to estimate the location. This estimation is based on the fingerprinting map, which represents a process of recording the RSSI together with the unique ID of each WAP that is detectable, and each corresponding location inside the building. Finally, the k-nearest neighbor search algorithm is used to estimate the location.

In this study, to take account for disaster scenarios in which there are more uncertainties in the RSSI values, the modified Naïve Bayes classifier is adopted instead of the k-nearest neighbor to provide the location based on fingerprinting data. Referring to the VPS flowchart shown in Figure 4, the RSSI signal is first collected by a smartphone, along with the Basic Service Set Identification (BSSID), which is the unique ID for each WAP. Appropriate features are then obtained by reading the RSSI. Finally, the Naïve Bayes classifier is applied to these features to estimate the most likely location of the victim. To consider a scenario in which some WAPs are not detected due to a disaster, the Naïve Bayes classifier was modified to consider only the features that are detected.

The location information can be displayed locally on the smartphone as part of a system to guide the victim to the nearest safe exit, and can be transmitted for use by on-site first responders.

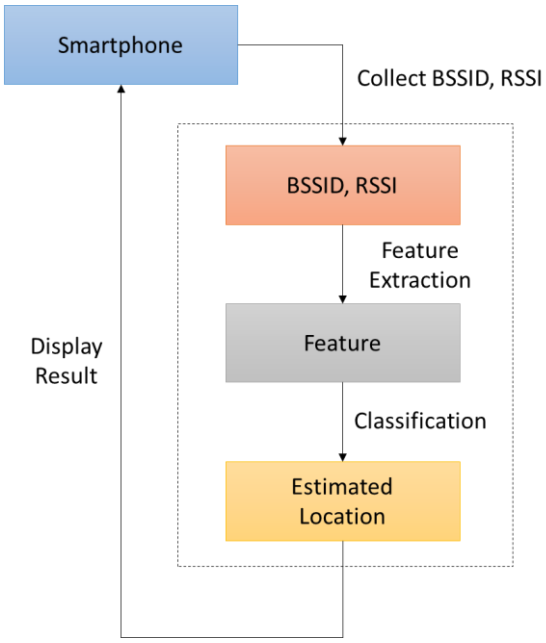


Figure 4: Flowchart of VPS.

Wi-Fi Received Signal Strength Indicator

Wi-Fi RSSI is one of the popular indicators used for indoor localization. The RSSI is a measurement of the power present in a radio signal received by an antenna. Assuming the WAP radiates a consistent Wi-Fi signal in free space without any obstacles, the RSSI is expected to be proportional to the inverse of the square of distance to the WAP. However, in reality, the radiated Wi-Fi signal can be reflected or refracted by fixed or moving objects (e.g., walls or people) and consequentially can exhibit spatial and temporal instability. This uncertainty prevents direct use of the trilateration technique to localize the phone by taking RSSIs from multiple APs (Mok and Retscher 2007).

Wi-Fi RSSI Scanner

To capture the RSSI of multiple WAPs, a Wi-Fi RSSI scanner module was developed and deployed on the Android OS smartphone platform, as shown in Figure 5. Scanning the WAPs does not require a connection to be made, but only to record the Service Set Identification (SSID), the Basic Service Set Identification (BSSID), and the corresponding RSSI for each detected WAP. SSID is the name of the Wi-Fi network, and BSSID is the Media Access Control (MAC) address of each WAP, each address being a unique identifier. Different colors are used to make each detected WAP distinguishable.

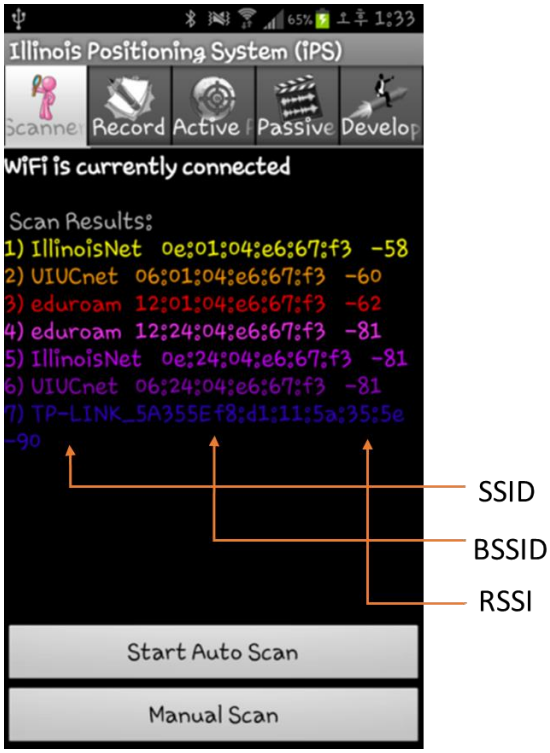


Figure 5: Screenshot for Wi-Fi RSSI scanner.

Characterization of RSSI Variability

At a fixed location in a building, the RSSI levels detected by a smartphone may vary over time depending on several factors. If this variability is too large it may negatively impact localization performance. I performed two experiments to assess RSSI variability in order to better understand the implications for our localization algorithm.

Effect of Phone Direction

One important factor influencing the RSSI level is the positioning of the user's body relative to the phone and the WAP signal source, as high water content in the human body attenuates or "shadows" the Wi-Fi signal (Bahl and Padmanabhan 2000). To characterize this effect, a test was conducted in a large room with a single WAP located at the center of the space. At three different distances from the WAP (i.e., 1m, 3m, 5m), the RSSI was measured with the user holding the phone facing directly towards or directly away from the WAP. Table 1 shows the average RSSI values for 10 measurements obtained from the test. Facing toward the WAP resulted in larger average RSSI values than when the user was facing away from the WAP for all three distances. The result shows that the user's body shadows the Wi-Fi signal, and the attenuation is greater when closer to the WAP. This effect of user body on the received signal strength is even more significant when the initial received signal strength from the WAP is weaker. This test indicates that the orientation of the user should be considered in the fingerprinting process to improve the accuracy of localization.

Table 1: Effects of body shadowing on measured RSSI (Higher values represent more powerful signals).

	1m		3m		5m	
Direction	Facing Towards	Facing Away	Facing Towards	Facing Away	Facing Towards	Facing Away
Avg RSSI (dB)	-32.9	-44.4	-36.7	-41	-49.7	-53.8
STD	1.22	3.26	1.84	1.10	1.10	2.68
Diff (dB)	-11.5		-4.3		-4.1	

RSSI Stability

Even with a fixed user orientation, RSSI levels measured by a phone at a fixed location were found to be variable over time. To characterize real-world variation in the RSSI, a test was carried out in a building on a college campus, with many students passing by in the vicinity of that building. The RSSI values for 10 detectable WAPs were measured at a specific location, with a 0.1 Hz sampling rate for 100 seconds. Figure 6 is a box-and-whisker plot (Tukey 1977) of the measured RSSIs for the 10 WAPs. The box-and-whisker plot shows the median, as well as the variation of the RSSI for each AP. On each box the center mark is the median; the edges of the box are the 25th and 75th percentiles; the whiskers extend to the most extreme data points except for outliers; and outliers are individually plotted with "+" marks. In these tests, the temporal fluctuation in the RSSI was usually within the range of about 5dB. Because, in most buildings, the RSSI value in different rooms varies by around 10 dB, the RSSI signal can still be used for localization even with a temporal variation of 5dB. This variation can be decreased by temporal averaging, which is used in the construction of the feature set for the Naive Bayes classifier, as described in Section 2.

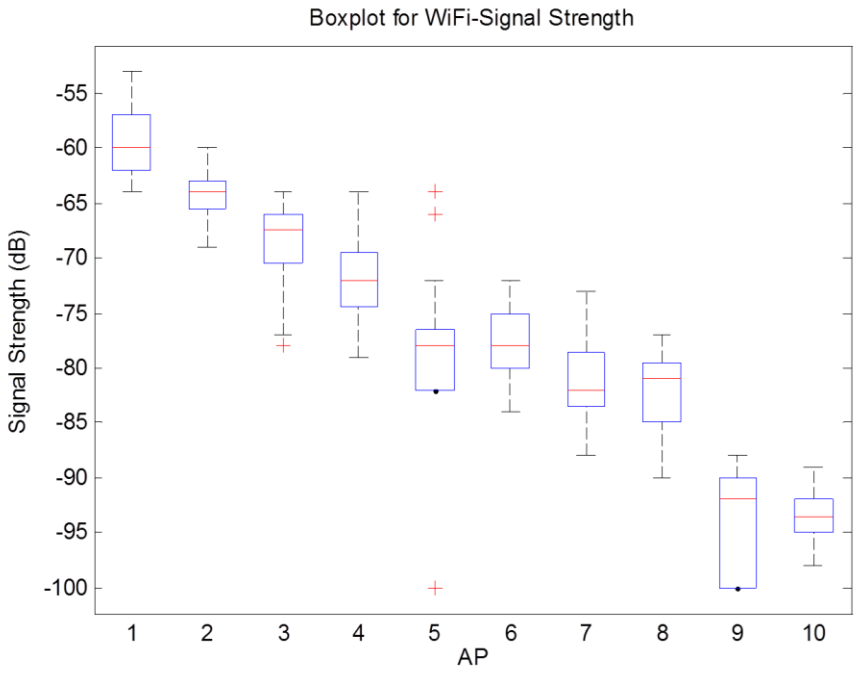


Figure 6: Box-and-whisker plot illustrating RSSI variability over a 100-second recording epoch for 10 different wireless access points detectable from a single location.

Localization Using RSSI

Before the fingerprinting step, a feature is developed from the measured RSSI values to localize the phone with improved accuracy. Two types of modifications were included in the development. First, the temporal variation of the measured RSSI, which was shown in Figure 6, is minimized by taking average of the RSSI values. The RSSI scanner module can sample RSSI every 0.1 seconds, and the average of ten consecutive RSSI values, measured for 1 second, is considered to be the representative RSSI at a location. Second, undetected WAPs are assigned a default RSSI value of -100 dB, which is the smallest value that an Android system can reliably detect.

Therefore, the RSSI feature is defined as,

$$F_i = \begin{cases} \overline{RSSI}_i & if\ \ \overline{RSSI}_i \geq -100\ dB \\ -100 & if\ \overline{RSSI}_i < -100\ dB\ or\ WAP_i\ is\ not\ detected \end{cases} \quad (6)$$

where $\overline{RSSI}_i$ is the average of ten consecutive RSSI values from the WAP_i.

Fingerprinting Using the Naïve Bayes Classifier

Fingerprinting is a process of developing a signal strength map using unique tags of WAPs for densely distributed locations in a building. This fingerprinting procedure is enabled by the Naïve Based Classifier, based on the localization feature shown in Equation (6). The goal is to reach room-level accuracy. For a specific room, the localization features are obtained at eight locations with different phone orientations, as shown in Figure 7. Several measurements are carried out to build conditional probability according to the Naïve Bayes Classifier.

The Naïve Bayes Classifier works based on the assumption that the features used in the Classifier contain the probability pattern of normal distribution. Under regular operational conditions, the RSSI usually follows the normal distribution (Bose and Foh 2007; Zanca et al. 2008). However, there is some research that shows that the RSSI values are not normally distributed, but rather, skewed (Kaemarungsi and Krishnamurthy 2004; Ladd et al. 2005). In the literature the skew is caused by users' interference. Thus, the Naïve Bayes Classifier can be used effectively if the direction effect is also included in the fingerprint mapping process.

The conditional probability for the certain feature F_i having $RSSI_i$ from the location l_j can be calculated from the PDF constructed by using the collected RSSI data (i.e., fingerprinted data). The distribution of RSSI for each WAP when the device is in a certain room is assumed as a Normal Distribution, as follows:

$$\mathrm{P}\left(F_i \middle| L = l_j\right) = \frac{1}{\sigma_{j,i}\sqrt{2\pi}} e^{-\frac{\left(f_i - \mu_{j,i}\right)^2}{2\sigma_{j,i}^2}} \tag{7}$$

If a signal is consistent for a certain location, the fingerprinting map can be constructed by using a single measurement at each location. However, as stated in the previous section, the Wi-Fi RSSI has uncertainty due to temporal instability and the directional effect. To minimize the temporal instability, the measurement was done several times in different time periods. In order to take account for the directionality issue, fingerprinting was done for eight different directions for each room, as shown in Figure 7. Each room r will contain eight location IDs (l) for each direction. (e.g., $r_j = [l_{8j-7}, l_{8j-6}, \dots, l_{8j}]$)

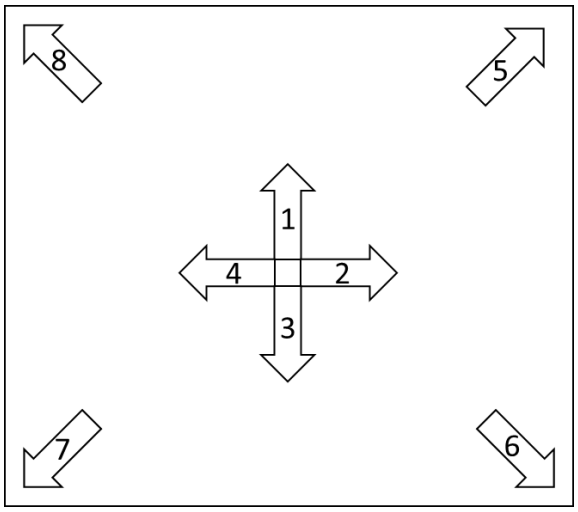


Figure 7: Fingerprinting of Room Level Considering the Phone Direction Effect.

Maximum Likelihood Method with Selective Features

Finally, the unknown location can be estimated by using the Maximum Likelihood Method, shown in Eq. (5). However, unlike the original Naïve Bayes Classifier, the number of features detected in

a room can be different from those detected in another room. This problem was resolved in the fingerprinting stage by setting the threshold values for features that had not been detected. However, when calculating the maximum likelihood, this approach will not only increase the calculation cost, but also will decrease the accuracy. In particular, in an emergency situation, some of the RSSI values might remain undetected because of damaged or unpowered WAPs. Therefore, the original classifier was modified to take into consideration only the features that have appeared in the test data: these will be considered in the equation below. The prior probability $P(R = r)$ was considered as a constant value without any additional information, and thus was neglected.

$$\text{Classify}\left(F'_1, \ldots, F'_m\right) = arg\max_r \prod_{i=1}^{m} \text{P}\left(F'_i \middle| R\right) \quad (8)$$

where F'_i is the re-numbered feature for the detected WAPs only, and m is the number of the detected WAPs.

3.2 Victim Assessment

Configuration of the Victim Assessment System (VAS)

The Victim Assessment System (VAS) is designed to assess and inform first responders of the status of victims, along with their locations estimated by the VPS. This information is necessary for preparing for evacuation plans and appropriate treatment of victims. The easiest and most accurate way to check the status of victims is by communicating directly with them, in the same way a doctor asks a patient about his condition. However, in disaster situations, phone lines may be down due to either physical damage to infrastructure or network congestion. Even if phone lines and internet network are available, first responders may not be able to communicate directly with all potential victims to determine their respective status within a reasonable amount of time. To

communicate with victims more efficiently, the VAS is composed of two sub-systems that use a local network which is assumed to have survived the disaster, or which can be established post-disaster by deploying an area of WAPS around the disaster site: (1) an Active Victim Assessment System (AVAS), a sub-system collecting useful information from conscious victims by the use of questionnaires, and (2) a Passive Victim Assessment System (PVAS), a sub-system continuously monitoring victims by using sensors inside their smartphones. AVAS will send questions directly to the victims, and PVAS will estimate the status of those who are not able to respond, or fail to respond to queries from the AVAS (Figure 8).

The Active Victim Assessment System (AVAS)

AVAS asks simple questions about the physical status of the victims inside a building, via smartphones. The questions can be selected by first responders according to the type of disaster or emergency. Questions can be answered simply by selecting either a “yes” or “no” button, so that even injured victims might be able to report their current situation to first responders. Figure 8 shows sample questions for AVAS. AVAS will determine whether the victims are ambulatory or non-ambulatory, based on their answers. Furthermore, AVAS will collect useful information that victims may want to report via voice messages. All of this information would be aggregated and made available to first responders on a timely basis.

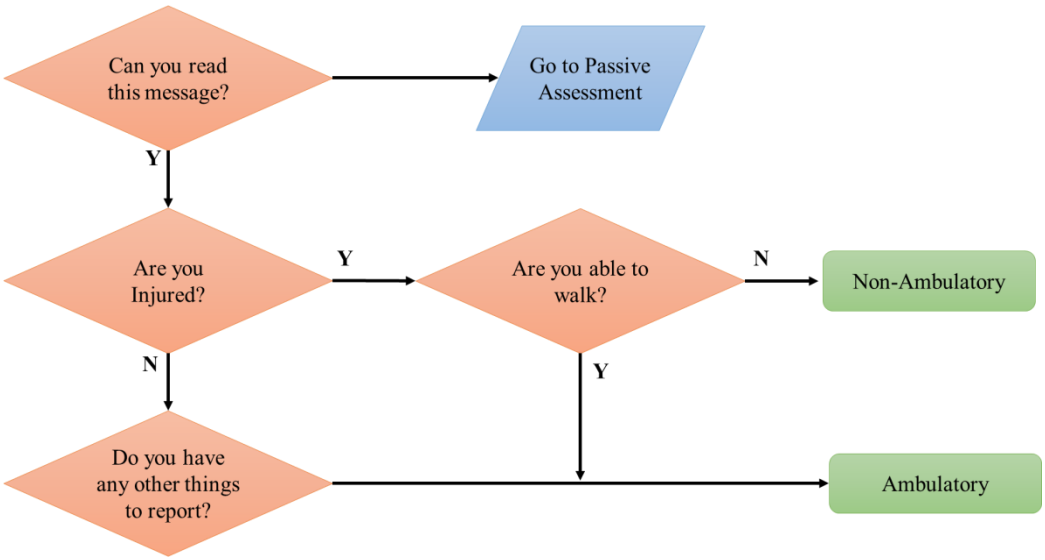


Figure 8: Flowchart with sample questions to determine victim status using AVAS.

The Passive Victim Assessment System (PVAS)

PVAS assesses the status of a victim in an automated manner by collecting real-time data from sensors, such as an accelerometer, a gyroscope, or a magnetic field sensor embedded in a smartphone. PVAS is developed for victims who are unconscious due to injury and who therefore are not aware of this application running on their smartphones. This system is enabled automatically if no response is collected by the AVAS, as shown in Figure 9. PVAS recognizes eight different types of activities (walking, running, standing, sitting/lying, rolling, fainting, walking upstairs, or walking down stairs) using the Naïve Bayes Classifier, with periodic updating. Estimated activities could be further linked to the four physical statuses of victims (highly ambulatory, ambulatory, non-ambulatory or unconscious) in order to aid first responders in coordinating evacuation and rescue efforts. Figure 9 shows the flowchart of determining the status of a victim while using PVAS. For best results, victims must have their smartphones in their

pocket—not in a bag—to capture victims' responses, using smartphones. The remaining portion of this section describes how PVAS works automatically during a given disaster.

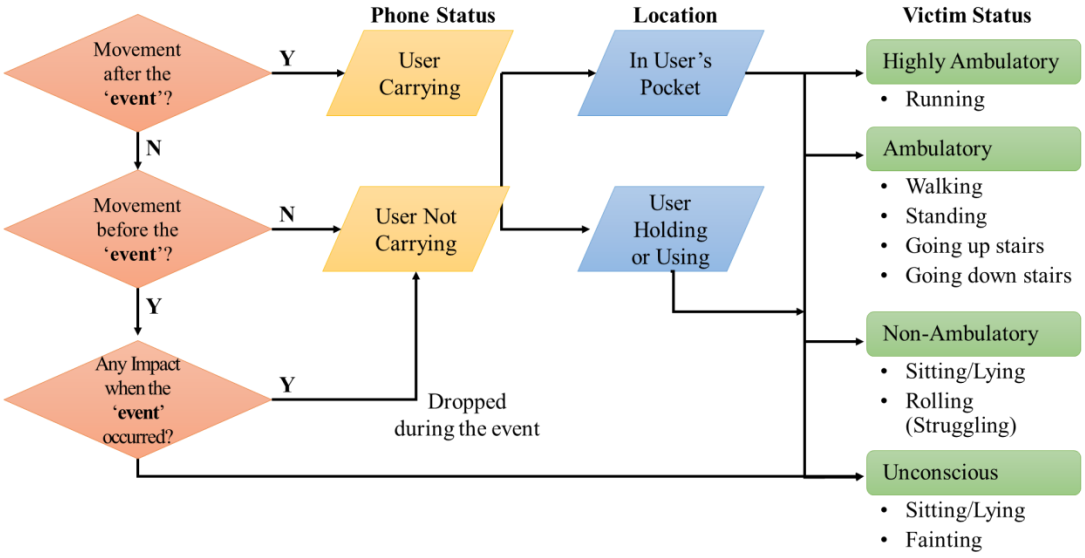


Figure 9: Flowchart for determining status, using PVAS.

Data Collection and Preprocessing

Sensor Data Scanner

To find the best features for assessing a victim's status using the sensor data experimentally, a sensor data scanner module is developed on the Android OS and embedded in the PVAS, as shown in Figure 10. The types of measurements and sampling frequency are adjustable by the user: 3-axis acceleration and 3-axis magnetic field data are collected at a rate of 10 Hz in this study. The orientation of the phone, calculated by the Android OS, is also displayed. The measurement can be exported into a text file for further development of the software.

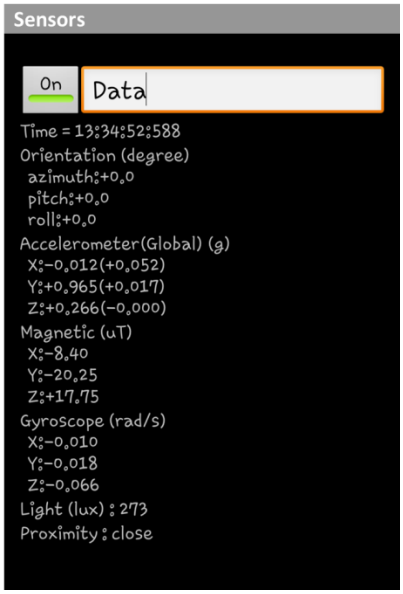


Figure 10: Screenshot for a Sensor Data Scanner.

Transforming Acceleration Sensor Data into a Global Coordinate System

The acceleration data of the phone is originally collected in the local coordinate system of the phone, as shown in Figure 11. The local coordinate system in the Android OS defines the x-axis as horizontal, the y-axis as vertical, and the z-axis as perpendicular to the phone. The global coordinate system defines the X-axis as west, the Y-axis as north, and the Z-axis as towards the center of the Earth. Acceleration in the local coordinate system (e.g. a_x, a_y, and a_z) varies according to the phone's orientation, and must be transformed into the global coordinate system (e.g. a_X, a_Y, and a_Z) that is invariant to the orientation of the phone.

In this study the transformation is made using the 3D transformation matrix (T), which is composed of three well-known Euler angles: roll (θ_x), pitch (θ_y), and yaw (θ_z) (Diebel 2006). The angles around the local axes can be obtained by using two conditions: 1) According to the global coordinate system, shown in Figure 11, the global acceleration will have zero values for X and Y directions and a constant of gravitation (i.e., 1g) for Z direction when there is no external

acceleration other than gravitation—and, the global magnetic field directed from south to north will always have zero value for X direction heading west. Next, the transformation matrix T can be obtained from

$$\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = T \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \tag{9}$$

$$\begin{bmatrix} 0 \\ m_{\mathrm{Y}} \\ m_{\mathrm{Z}} \end{bmatrix} = T \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} \tag{10}$$

where a and m denote the measured acceleration and magnetic field, respectively; their small and large subscripts denote local and global coordinate systems, respectively. Then, global acceleration can be obtained using the transformation matrix as

$$\begin{bmatrix} a_X \\ a_Y \\ a_Z \end{bmatrix} = T \begin{bmatrix} a_z \\ a_y \\ a_z \end{bmatrix} \tag{11}$$

Note that, when θ_y is $90°$, i.e. $cos\theta_y = 0$, the Gimbal Lock problem may occur (King 1998). Quaternions may provide the solution for avoiding this problem (Kuipers 1999).

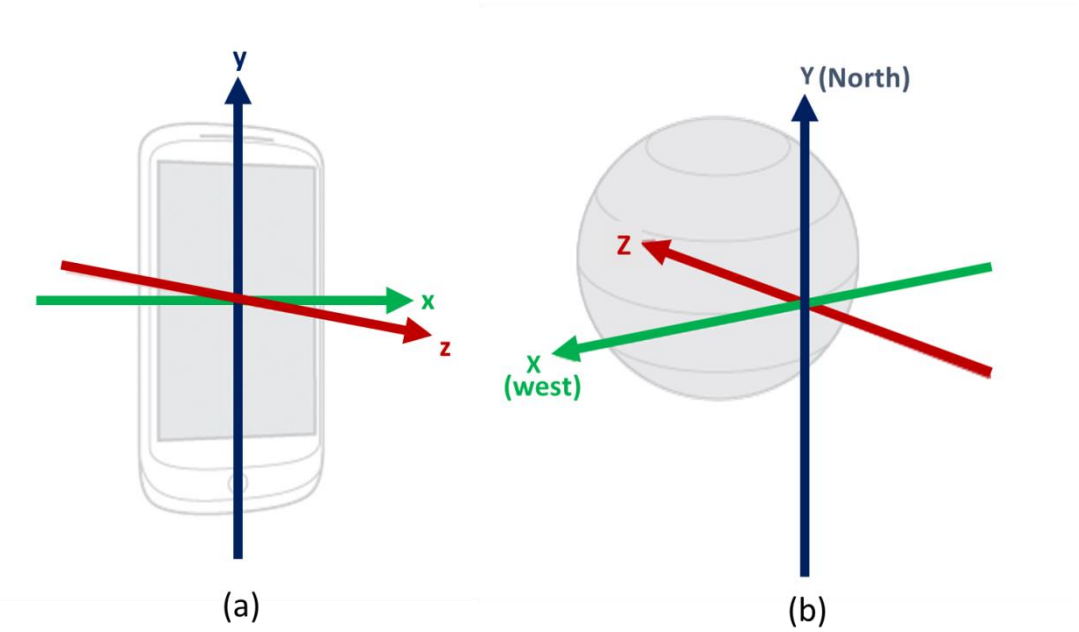


Figure 11: (a) Local Coordinate System, and (b) Global Coordinate System.

Even though the coordinate transformation of the sensor data from the device that is translated into global coordinates has been introduced, the system still relies upon the assumption that a smartphone is either in use (Active VAS) or in one's pocket (Passive VAS). These two components of the VAS will cover a substantial portion of the victims trapped in the building, but those who do not have their smartphones with them or who keep their phones in their bags or purses will not receive the full benefit of the system. The system can be expanded to more general cases by using wearable devices such as smart watches or glasses, so that in the future more people can receive the benefits of the system.

Features for a Victim's Status

Appropriate processing of acceleration data yields features that can assess the status of victims by using pattern classification methods, such as the Naïve Bayes Classifier. In this study six

distinguishable features have been selected to identify eight activities, based on an intuitive understanding of human activities, as summarized in Table 2. These features have been extracted from an acceleration in the global coordinate system and magnetic field. Absolute vertical and horizontal acceleration ($\mathbf{F_1}$ and $\mathbf{F_2}$) distinguish movements with vigorous vertical (e.g., running, walking) or horizontal (e.g., running, walking, rolling) vibration from static movements. Movements that are more likely to be horizontal (e.g., rolling) can be distinguished by using the ratio between the vertical and horizontal acceleration levels ($\mathbf{F_3}$). The variance of velocity ($\mathbf{F_4}$) distinguishes movements with high variance (e.g., fainting) from low variance (e.g., standing). Stationary movement with high frequency (e.g. running) can be distinguished from movement with low frequency (e.g. walking) by using dominant frequency ($\mathbf{F_5}$). Finally, the orientation of a phone ($\mathbf{F_6}$) can distinguish whether the phone is in a position (i.e., laying/sitting) parallel to the floor or perpendicular to it (e.g., standing).

The flowchart for the feature extraction is shown in Figure 12. The DC offset of acceleration, assumed to correspond to the gravitation, is calculated, and the orientation of the phone is calculated using the DC offset and magnetic field data. The acceleration data in the local coordinate system, after removing the DC offsets, is then transformed into a global coordinate system using the estimated orientation. The features $\mathbf{F_1} \sim \mathbf{F_4}$ are calculated from the global acceleration, and $\mathbf{F_5}$ is obtained from the Fourier spectrum of measured acceleration. F_6 contains two components (i.e., pitch and roll) of the phone. The Fourier spectrum is calculated using 64 acceleration data so that $\mathbf{F_5}$ is updated every 6.4 seconds. The other features are also updated every 6.4 seconds.

Table 2: Features for PVAS.

Feature	Mathematical Expression	Distinguishable Activities		
		Large	**Moderate**	**Small**
$\boldsymbol{F_1}$	$\sqrt{\mathbf{E}[\boldsymbol{a_Z^2}]}$	Running	Walking	Standing Lying/Sitting
$\boldsymbol{F_2}$	$\sqrt{\mathbf{E}[\boldsymbol{a_X^2} + \boldsymbol{a_Y^2}]}$	Rolling	Running Walking	Standing Lying/Sitting
$\boldsymbol{F_3}$	$\frac{\sqrt{\mathbf{E}[\boldsymbol{a_Z^2}]}}{\sqrt{\mathbf{E}[\boldsymbol{a_X^2} + \boldsymbol{a_Y^2}]}}$	Fainting	Running Walking	Rolling
$\boldsymbol{F_4}$	$\mathbf{Var}[\boldsymbol{v_Z}]$	Fainting	Running, Walking	Standing Laying/Sitting
$\boldsymbol{F_5}$	$\mathbf{argmax}_{\boldsymbol{f}}(\mathcal{F}\ (\boldsymbol{a_Z}))$	Running		Walking
$\boldsymbol{F_6}$	$\mathbf{abs}(\boldsymbol{\theta_X} + \boldsymbol{\theta_Y})$	Standing		Lying/Sitting

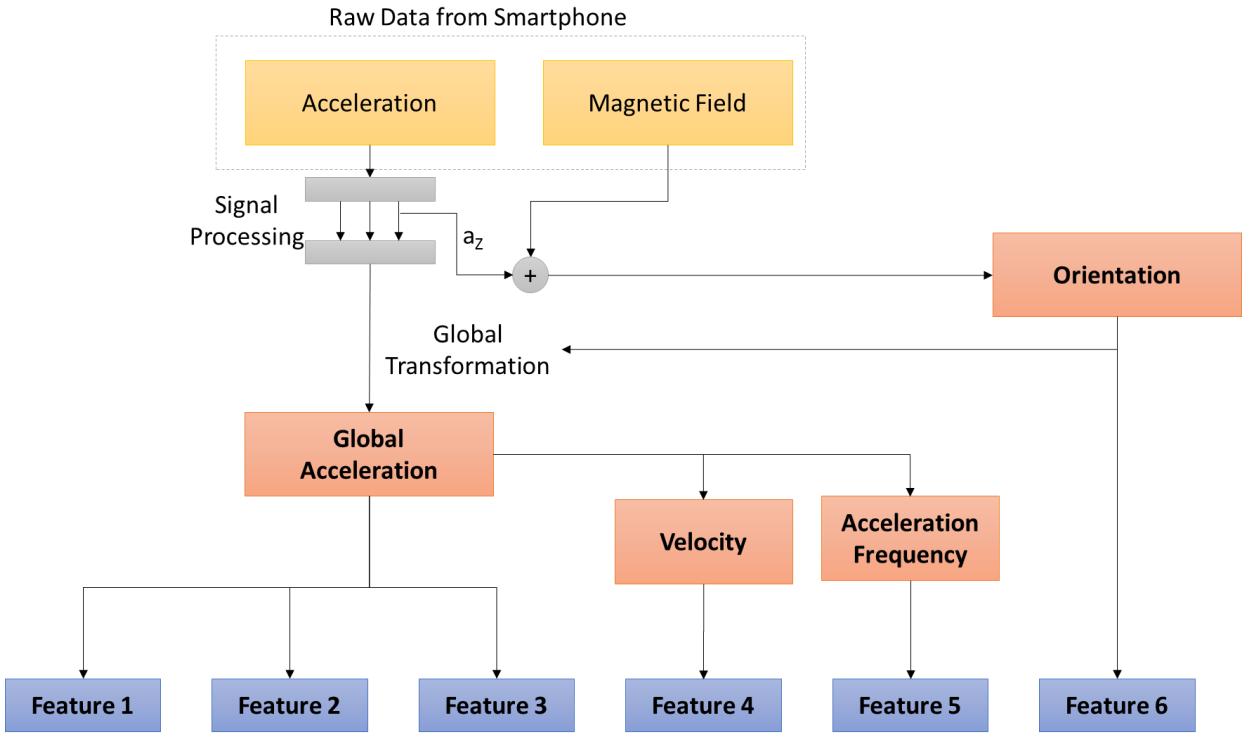


Figure 12: Flowchart for feature extraction from sensor data.

Status Assessment Using the Naïve Bayes Classifier

After getting the features, the activities of the victim are obtained using the Naïve Bayes Classifier; these activities can be used to infer the status of a victim. The probability of a data set having features $F_1, F_2, \ldots, F_n$ being class C_i can be calculated with Eq. (4). The success rate of the classification by the Naïve Bayes Classifier significantly depends on the type of probability density function inherent in the features. To select the appropriate distribution for each feature, a normality test was done for the features obtained from the sampled measurements for each activity. Figure 13 shows the sample normal and log-normal probability plots of F_1. The '+' symbol, which indicates the sample data, is plotted with a red line, indicating a robust linear fit of the sample. If the sample data fits into the distribution exactly, the blue '+' symbols will show a linear line. The other features, $F_2 \sim F_6$, also showed a similar pattern, which follows the log-normal distribution,

although they are not shown in this work. While a normal distribution was used to represent the features for VPS (e.g., RSSI), the features for PVAS, on the other hand, do not follow the normal distribution. The result indicates that the features in PVAS are better represented by a log-normal distribution. Therefore, the conditional probability for the particular feature F_i for activity status s_j can be calculated from the PDF as constructed by the Log-Normal Distribution, as shown below.

$$\mathrm{P}(F_i|S = s_j) = \frac{1}{\sigma_{j,i}\sqrt{2\pi}} e^{-\frac{\left(logf_i - \mu_{j,i}\right)^2}{2\sigma_{j,i}^2}}$$

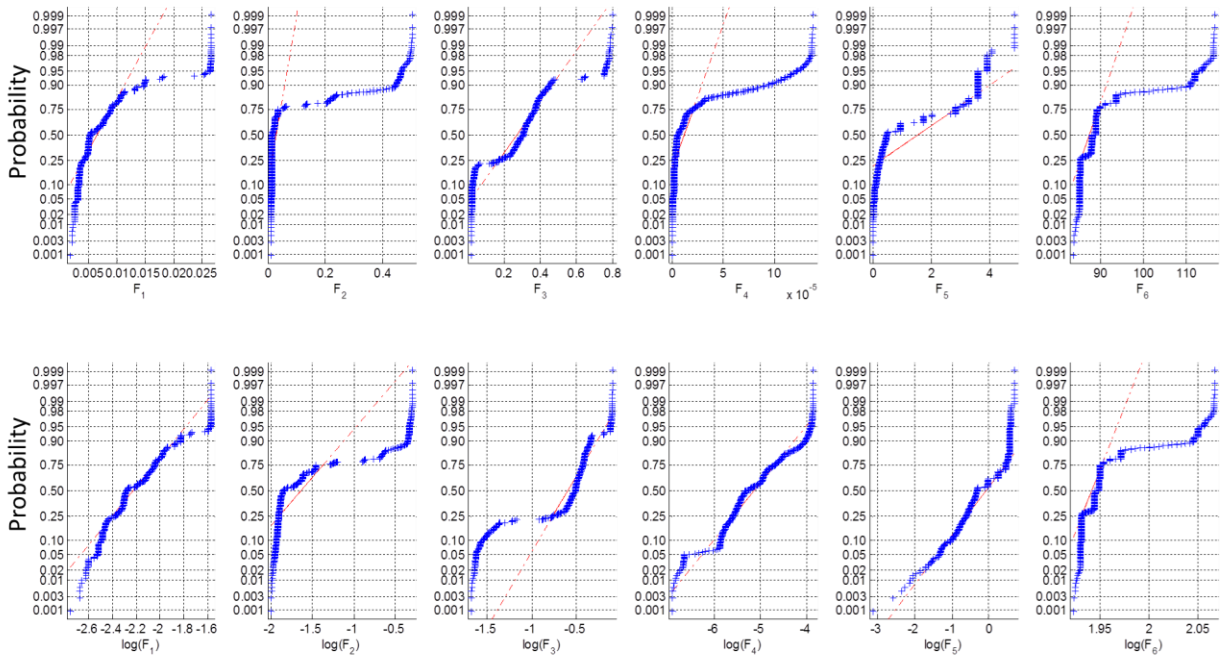


Figure 13: Probability Plot for Features 1~6 (top: normal, bottom: log-normal).

3.3 Integrated Victim Rescue System: iRescue

iRescue, an Android-based victim rescue system, was developed by integrating the two previously explained systems, VPS and VAS, as shown in Figure 14. The current version of iRescue has three tabs for verifying the performance of the proposed VPS and VAS. The VPS tab (Figure 14.b)

estimates the current location of the user and shows that location on the map. The AVAS tab (Figure 14.c) presents questionnaires directly to the user to obtain a response. The PVAS tab (Figure 14.d) estimates the status of the victim and visualizes the results on the phone. All of this information (the estimated location and status of the victim) will be transmitted to the first responders in real time via a TCP-IP protocol, using the local network. The current version of the application does not have ability to be installed and launch automatically when an emergency occurs. However, if the proposed system is adopted for practical use, this issue can potentially be solved by recommending wireless carriers pre-install the application prior to activation. Finally, when an emergency occurs, it is assumed that the application is launched remotely by first responders.

Figure 14: Screenshots for the iRescue (a) startup screen, (b) VPS, (c) AVAS and (d) PVAS.

VPS calculates the probability of being in a certain location (Figure 15, left) and shows the location with the highest probability, together with a floor map that contains a touch interface

(Figure 15, right). The estimated location will be shown at the top when pressing the "locate" button at the top left corner of the screen. It will also indicate the current location on the map using the blue circle. VAS calculates the probabilities of being in a certain status (Figure 16) and shows the activity having the highest probability within every 6.4 seconds. Also, the recent activity history will be displayed by images that could be used to determine the status of the user.

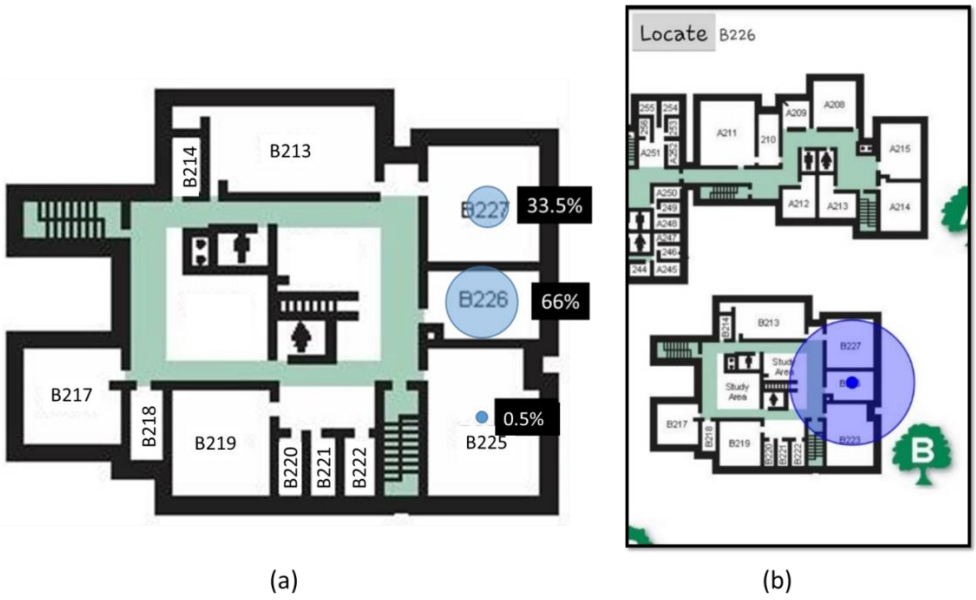


Figure 15: (a) P-values of VPS for sample data at B226, and (b) final result for VPS

Figure 16: Activities for VAS.

3.4 Validation Test

System Validation

The performance of *iRescue* for the localization and assessment of victims in a building has been investigated by means of a series of validation tests. In this section, the validation tests for the VPS and VAS have been carried out separately. The validation test for the VPS was carried out on the 2nd floor of Wing B of Parkland College, located in Champaign, Illinois. The validation of the VAS was conducted in two ways: activity simulation in a building with five people and the real-time monitoring of a person equipped with various devices that were recording his activities. The smartphone used in the validation tests is a basic "HTC Nexus One".

Validation Test: The Victim Positioning System

The validation test of the developed VPS was conducted on the 2nd floor of Wing B of Parkland College, located in Champaign, Illinois. The map of the test area is shown in Figure 17. The test area has nine rooms and 34 detectable WAPs around the floor. For the validation, the RSSIs from the WAPs were collected for eight different directions, as shown in Figure 6. A 4-fold cross validation method was employed, which uses each quarter of data as testing and the other three-quarters as training (i.e., fingerprinting), as shown in Figure 18. Because the positioning does not depend on people's activity patterns, all the data was collected by a single person. The data was collected thirty times in three weeks to check for temporal variations in the building environment.

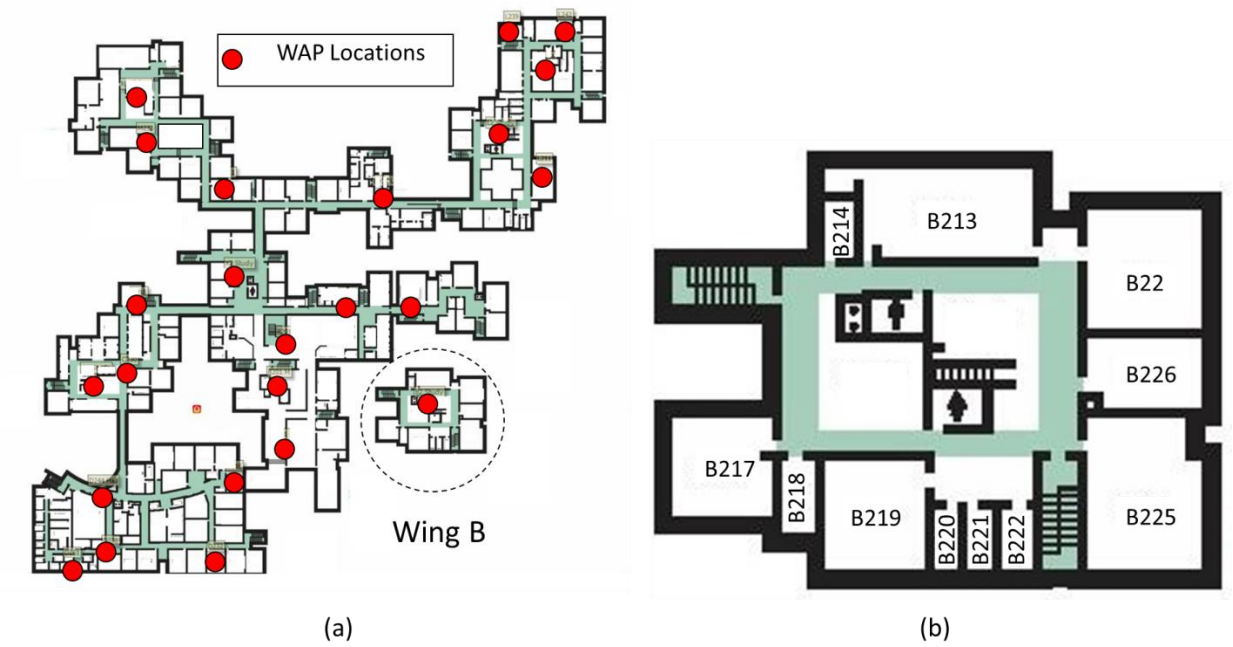


Figure 17: Floor plan and location of WAPs for the (a) 2nd floor of Parkland College, and (b) for Wing B.

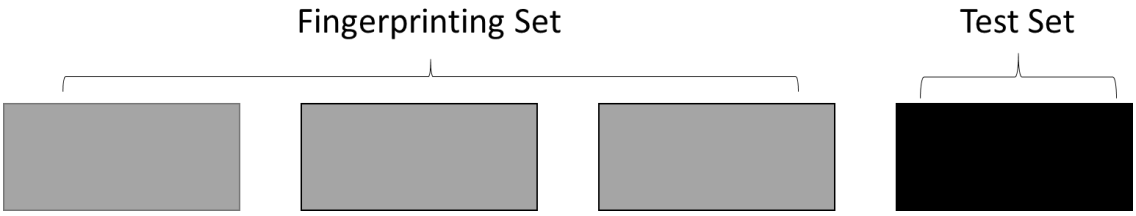


Figure 18: 4-Fold Cross Validation.

Result comparison for Naïve Bayes and the Modified Naïve Bays Classifier

The average success rate for using the original Naïve Bayes Classifier was about 72% in room-level accuracy, while the Modified NB that considered only the features that had appeared, was about 79%. Therefore, we see that the modified method can increase classification accuracy by 7% compared to the original Naïve Bayes Classifier (Figure 19).

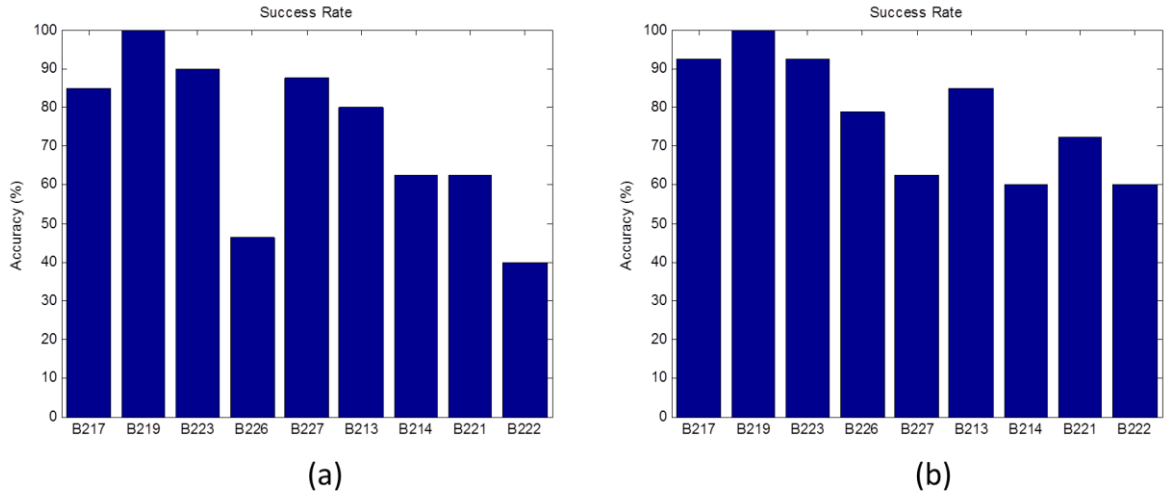


Figure 19: Success rate for using Naïve Bayes (a) and the Modified Naïve Bayes (b) Classifier.

Result comparison for the with and without-considering the directionality effects

Table 3 is a confusion matrix (Kohavi and Provost 1998) for the case without considering the directionality effect. Locations for eight different directions were labeled as all different classes for the data for the test considering the directionality effect, while locations in the same room were all labeled as same class for the test without considering the directionality. Each column of the confusion matrix represents the instances in a predicted class, while each row represents the instances in an actual class. Because the directionality of the device can have different RSSI values for certain WAPs, the classifier incorrectly identified some locations. For example, the RSSI from some WAPs for room B214 were more likely to be those of the next door which is B213 which led to this misclassification.

Figure 20 shows the classification result with and without considering the directionality effect. When considering the directionality, data obtained from different directions (Figure 4) in a single room was considered as different classes. In contrast, all of the directions in a single room

were considered as a single class for the latter case. When directionality was considered, some of this misclassification that was made in Table 3 was removed, which eventually increased the accuracy into 87%.

Table 3: Confusion Matrix without considering the directionality effect.

Actual Room	Localized Room								
	217	219	223	226	227	213	214	221	222
217	**.93**	.07	.00	.00	.00	.00	.00	.00	.00
219	.00	**1.00**	.00	.00	.00	.00	.00	.00	.00
223	.00	.00	**.93**	.07	.00	.00	.00	.00	.00
226	.00	.00	.00	**.79**	.21	.00	.00	.00	.00
227	.00	.00	.00	.37	**.63**	.00	.00	.00	.00
213	.00	.00	.00	.00	.00	**.85**	.15	.00	.00
214	.00	.00	.00	.00	.00	.40	**.60**	.00	.00
221	.00	.24	.00	.00	.00	.00	.00	**.73**	.03
222	.00	.00	.00	.00	.00	.00	.00	.40	**.60**

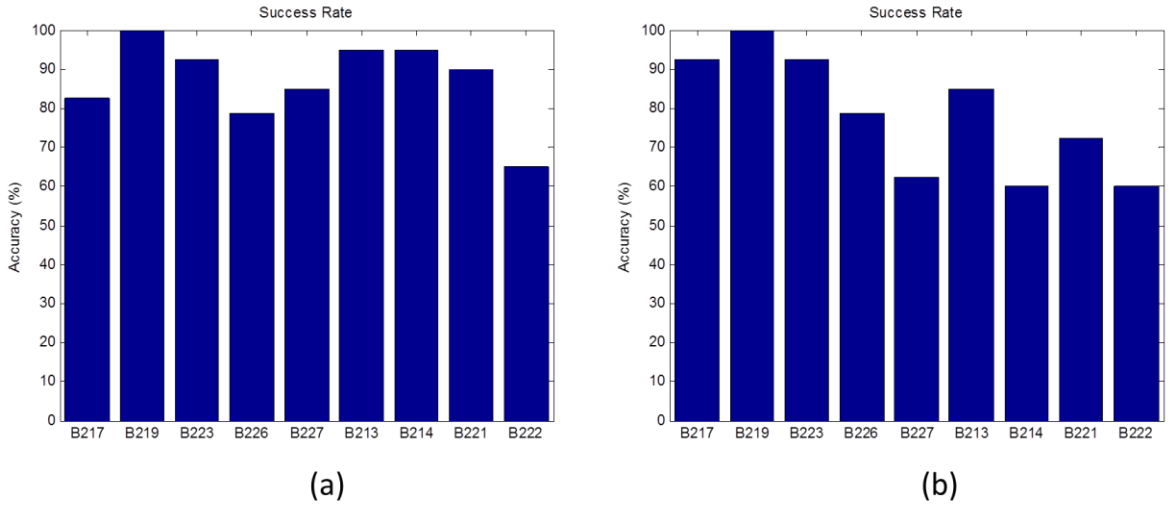


Figure 20: Success rate for (a) considering directionality, and (b) without considering directionality.

Victim Assessment System Validation

Two validation tests were conducted to verify the performance of the developed PVAS: activity simulation test with five participants and real-time monitoring of a person equipped with an activity-recording device.

Activity Simulation Test

The first test is an activity simulation test with five participants (A, B, C, D, and E), who kept a smartphone in their respective pants pockets. The participants were asked to simulate eight activities in their own natural ways. For each activity, the acceleration, magnetic field and orientation (calculated using the gravitation and magnetic field) were obtained for 60 seconds with a sampling rate of 10 Hz. Then, the first 300 data points for each person were used for training, and the other 300 for testing.

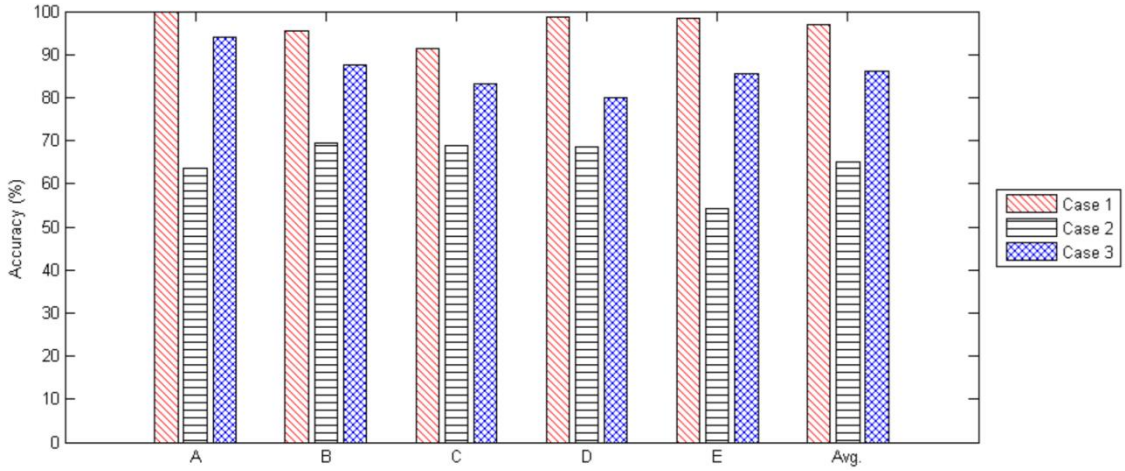


Figure 21: Accuracy of the VAS for each participant.

The result of the test is shown in Figure 21 by having three different cases of the training data to check the effect of the participants' unique activity patterns; Case 1, training data from the same participant; Case 2, from one of the other participants; and Case 3, from all other participants. When the trained data from the same participant was used (Case 1), the accuracy is very high (96.8%) on average—even 100% for Participant A. However, the accuracy decreases to 67.8% on average when the training data was taken from the other participant (Case 2). Since the training data for all people cannot be obtained for training prior to the disaster, the latter case would be more realistic. When the training data was obtained from other four participants (Case 3), the success rate significantly improved—up to 81% on average. This improvement shows that the data from a person displaying a similar activity pattern increases the accuracy of VAS, and that a reasonable success rate can be achieved by using the data from more people (for training).

A result is also obtained for each activity. Table 4 shows a confusion matrix from the activity simulation test. The assessed activities fully agree with the actual one for the catetories lay/sit, run, roll and faint. However, going up the stairs and going down the stairs are assessed with

low accuracy, since the two activities are actually combined with running or walking. From Table 4, the “Stair Up” was mostly misclassified as “Lay/Sit” or “Walk”, and “Stair Down” was mostly misclassified as “Run”. This misclassification could happen because most people tend to walk faster, and the act of stepping down induces a high vertical acceleration level, which is the most dominant feature classifying “Running”. Meanwhile, people tend to be slow and more stable when stepping upward. This tendency makes for a lesser acceleration level, which is the dominant feature classifying “Lay/Sit” and “Walk”. Going up the stairs and going down the stairs would be classified more accurately by using pressure sensors (barometer) in future work, which is now embedded in most of the newest smartphones.

Table 4: Confusion matrix for the Activity Simulation Test.

Actual Activities	Assessed Activities							
	Lay/sit	Stand	Walk	Run	Stair Up	Stair Down	Roll	Faint
Lay/sit	1.00	.00	.00	.00	.00	.00	.00	.00
Stand	.05	.87	.00	.00	.00	.00	.08	.00
Walk	.00	.00	.88	.12	.00	.00	.00	.00
Run	.00	.00	.00	1.00	.00	.00	.00	.00
Stair Up	.50	.00	.14	.00	.36	.00	.00	.00
Stair Down	.00	.00	.00	.23	.00	.77	.00	.00
Roll	.00	.00	.00	.00	.00	.00	1.00	.00
Faint	.00	.00	.00	.00	.00	.00	.00	1.00

Real-time Monitoring

While the first validation test was carried out to check for stationary results, an additional real-time monitoring test was conducted in order to validate the actual activity of the user. In the real-time monitoring test, the PVAS system was running in the phone with other equipment attached to the body: an automatic camera and a commercial health monitoring device called *SenseWear*, were attached to an armband that could estimate the energy expenditure of the user. As shown in Figure 22, a camera was hung on the user's neck and set to take a photo automatically every second in order to record every activity of the user during the test. *SenseWear* was attached to the user's arm to compare the estimated activity by PVAS with the energy expenditure estimation. The energy expenditure were expected to be high when performing an active activity (e.g., walking) and to be low when doing a static activity (e.g., standing).

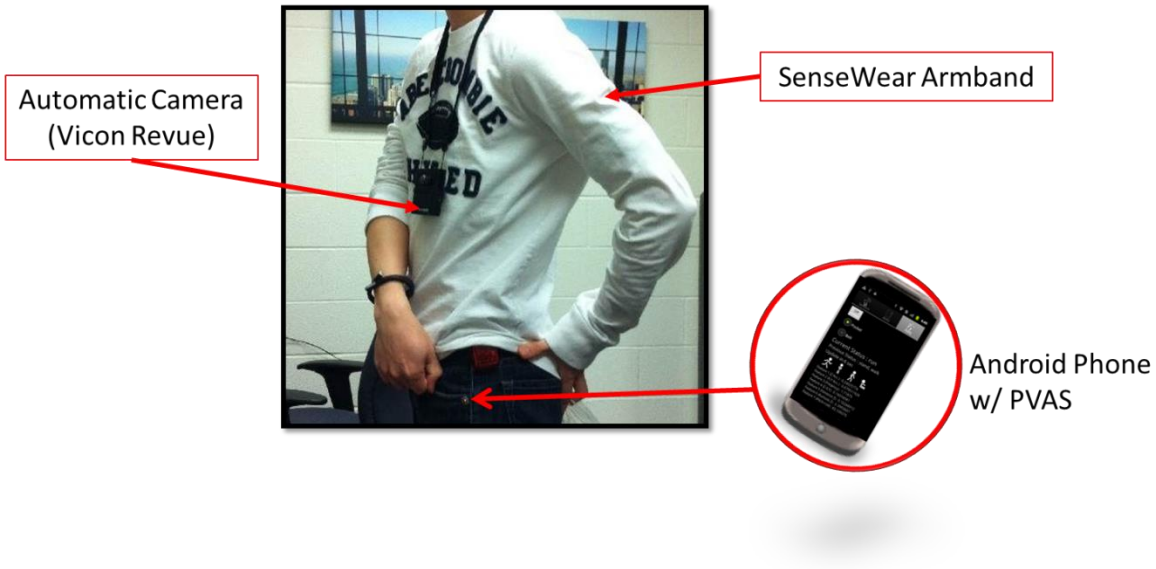


Figure 22: Configuration of equipment for real-time monitoring test.

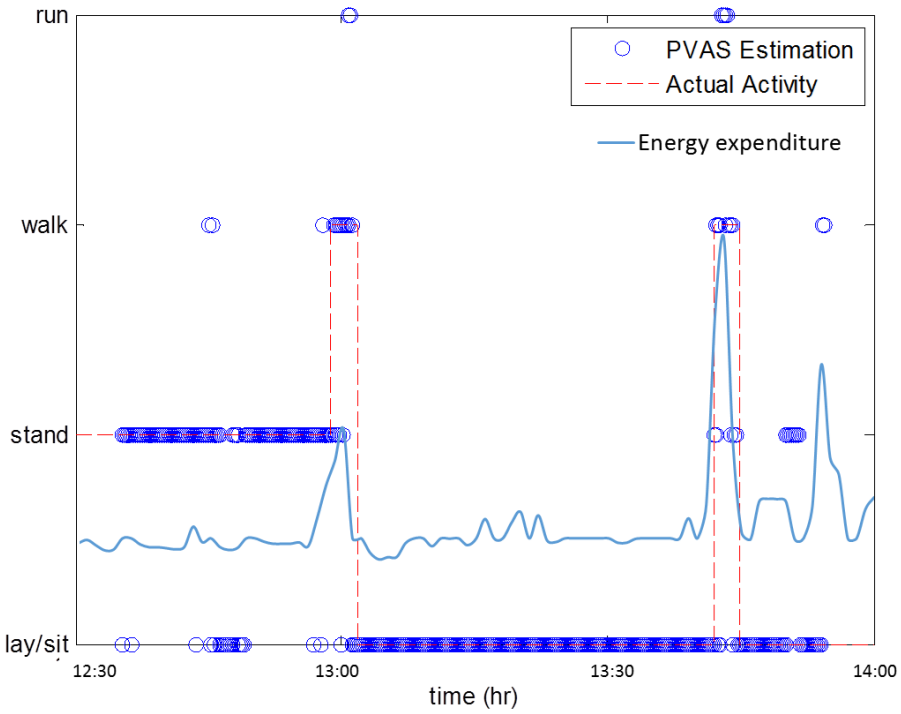


Figure 23: Comparison of assessed activities with actual activities and energy expenditure during the monitoring test.

The validation test was conducted for about 90 minutes during daylight hours. As shown in Figure 23, the PVAS estimated the first half hour as standing where the photo from the automatic camera indicated the person was standing. Also, the PVAS estimated the acvitivity of the next few minutes as walking, when the person was, in fact, walking for that period. The energy expenditure from the sensors attached on the person's body showed higher values when the PVAS estimated the activity as walking compared to when the person was standing. This result also supports the assertion that the PVAS system is correctly estimating the activity.

3.5 Summary

This task presented a new system to aid first responders' plans for evacuation and assistance of victims trapped inside a building. The task was based on the sensing and communication capabilities of victims' smartphones. The system estimates the location and physical status of victims inside a building by combining two developed sub-systems.

The Victim Positioning System (VPS) estimates the location of the user inside the building by using the received Wi-Fi RSSI by use of the phone. Wi-Fi RSSI was collected and has been analyzed in order to discover the characteristic of the signal. The distance-signal relationship test, directionality test and stability test were conducted in order to improve the accuracy of the classification. The modified Naïve Bayes Classifier was suggested so that the only selected features could be used for the classification. The fingerprinting was done at Parkland College in order to determine the performance of the system. The system was able to successfully locate victims at room-level inside a typical building with an accuracy of 87%.

The Victim Assessment System (VAS) was able to estimate the activity of the user by using acceleration and the magnetic field sensor of the phone. The sensor data were collected and have been analyzed to identify the characteristics of data for different activities. These characteristics were then used to select the features for the probabilistic classification known as the Naïve Bayes Classifier. The individual activity test showed an average accuracy of 81% by having training data for five different persons. Also, the real-time monitoring test, which compared the actual activity recorded by the camera, as well as other sensors, showed that the estimated status of the victim was, for the most part, correct.

CHAPTER 4 FIRST RESPONDER TRACKING AND EVENT LOCALIZATION

This chapter presents a new approach for tracking first responders and localizing events in both indoor and outdoor GPS-denied environments. Here, event localization refers to "finding the location of a user when conducting any value-added activities in emergency response", such as locations of the structural hazards, fires, or a shooter in ag` building. To do so, the proposed method leverages existing sensors that are embedded in a commodity smartphone and does not require the user to carry additional hardware. Using data obtained from these sensors, multiple 3D dead-reckoning paths are integrated in a probabilistic model. This method hypothesizes that the probabilistic integration of localization results from multiple dead-reckoning processes based on different reference points will produce more reliable localization results, as compared to single best estimations from a one-way dead-reckoning process. In the following, the proposed probabilistic approach for 3D event localization and the underlying algorithms using embedded sensors in a smartphone are presented. Then, the experimental results based on several case studies done about a multistory parking garage are presented—findings that validate our hypothesis. Perceived benefits and open research challenges are also discussed in detail. The validation test for the emergency response scenario remains for future work, and will be further discussed in Chapter 5.

4.1 Probabilistic Pedestrian Dead Reckoning

Building upon the work of Tian et al. (2014), the pedestrian dead reckoning mechanism tracks the motion trajectory of a mobile user by detecting the footsteps made by the mobile user and by

estimating the direction in which that person is heading ($\vec{\boldsymbol{h}}$) at any point in time. The location of the mobile user at step *i*+1 ($\boldsymbol{x_{i+1}}$) can be described by using the formerly determined position at step *i* ($\boldsymbol{x_i}$) and the stride length of the mobile user (L_s), as follows:

$$\boldsymbol{x_{i+1}} = \boldsymbol{x_i} + \vec{\boldsymbol{h}}_{\boldsymbol{i}}(L_s) \tag{12}$$

Thus, the final location of the mobile user with respect to the known landmark location can be described by using the following equation:

$$\boldsymbol{x_p} = \boldsymbol{x_{LM}} + \sum_{i=1}^{n_{step}} \vec{\boldsymbol{h}}_{\boldsymbol{i}}(L_s), \tag{13}$$

where $\boldsymbol{x_p}$ is the mobile user's location, $\boldsymbol{x_{LM}}$ is the landmark location, and n_{step} is the total number of steps detected from a smartphone. The pedestrian dead reckoning mechanism is composed of the following three underlying components (Figure 24).

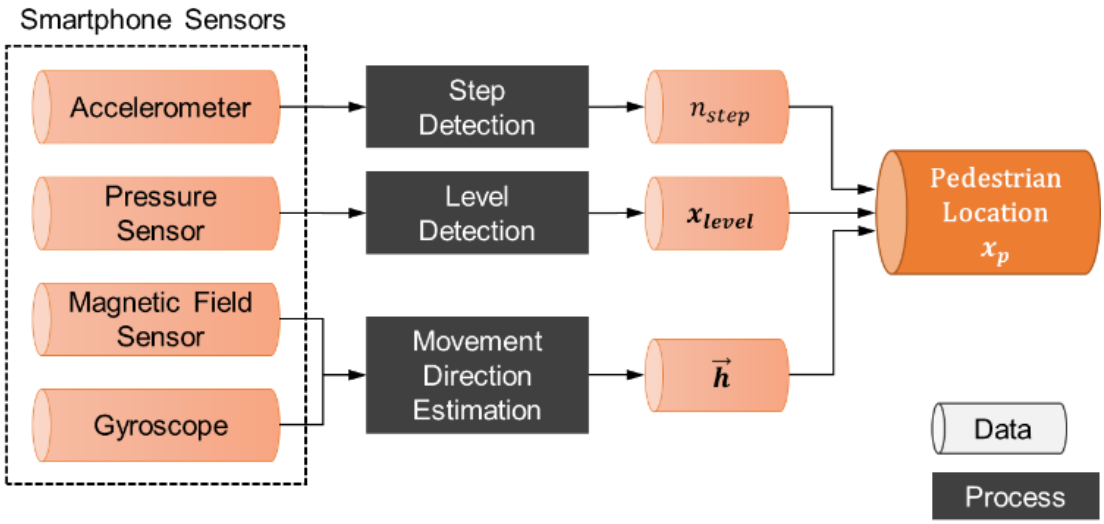


Figure 24: Data and process for the Pedestrian Dead Reckoning Mechanism.

(1) Step detection: The footstep of a mobile user can be detected by using the vertical acceleration data obtained from a smartphone. First, the low-pass filter is used to eliminate noise in the vertical acceleration data. Then, negative and positive peaks are detected in the distribution, and their difference is calculated. Finally, based on the predefined threshold, the footsteps are detected.

(2) Movement direction estimation: The movement direction at each time is estimated by using the gyroscope and magnetic field sensor. The magnetic field sensor is used to calculate one's initial direction ($\vec{\boldsymbol{h}}_{\mathbf{0}}$), and the gyroscope is used to correct the subsequent directions by cumulating the angular displacement ($\Delta\vec{\boldsymbol{h}}$) when the user changes the movement direction. The angular displacement ($\Delta\vec{\boldsymbol{h}}$) is obtained from first applying the low-pass filter to the vertical gyroscope data ($\boldsymbol{g}_{\boldsymbol{z}}$) for eliminating the noise, and then numerically integrating the data (Equation 16).

$$\vec{\boldsymbol{h}}_{\boldsymbol{i}} = \vec{\boldsymbol{h}}_{\boldsymbol{0}} + \sum \Delta\vec{\boldsymbol{h}} = \vec{\boldsymbol{h}}_{\boldsymbol{0}} + \int \boldsymbol{g}_{\boldsymbol{z}}\, \boldsymbol{dt} \tag{14}$$

(3) Level detection: In addition to the movement on a single floor, the level detection module estimates the building level where one is located for 3D dead reckoning. The elevation of a user is determined based on the pressure sensor readings in a smartphone. Typically, the pressure drops in proportion to the height. However, because the pressure sensor readings typically vary over time, deriving the building level solely based on the pressure sensor readings can be problematic. To address this issue, the difference in the pressure sensor readings at different times is used to estimate the building levels where the user is located. To do that, the pressure sensor readings are recorded over a certain time period and are compared with previously recorded values. If the difference between these two values (ΔP) is greater than the predefined threshold, the building level of the user is considered "changed".

4.2 Probabilistic Vehicle Dead Reckoning

Vehicle Dead Reckoning tracks a moving vehicle based on the vehicle's speed and the direction in which it is headed ($\vec{\boldsymbol{h}}$) for each time step. The vehicle location at time t+1 ($\boldsymbol{x_{t+1}}$) can be described by using the previous location at time t ($\boldsymbol{x_t}$) and the vehicle speed (v_t) as follows.

$$\boldsymbol{x_{t+1}} = \boldsymbol{x_t} + \vec{\boldsymbol{h}}(v_t)\Delta t \tag{15}$$

The vehicle location ($\boldsymbol{x_v}$) with respect to the certain turn location ($\boldsymbol{x_0}$) is described by following equation:

$$\boldsymbol{x_v} = \boldsymbol{x_0} + \sum_t \vec{\boldsymbol{h}}(v_t)\Delta t \tag{16}$$

The Vehicle Dead Reckoning example shown in Figure 25 is composed of the following modules:

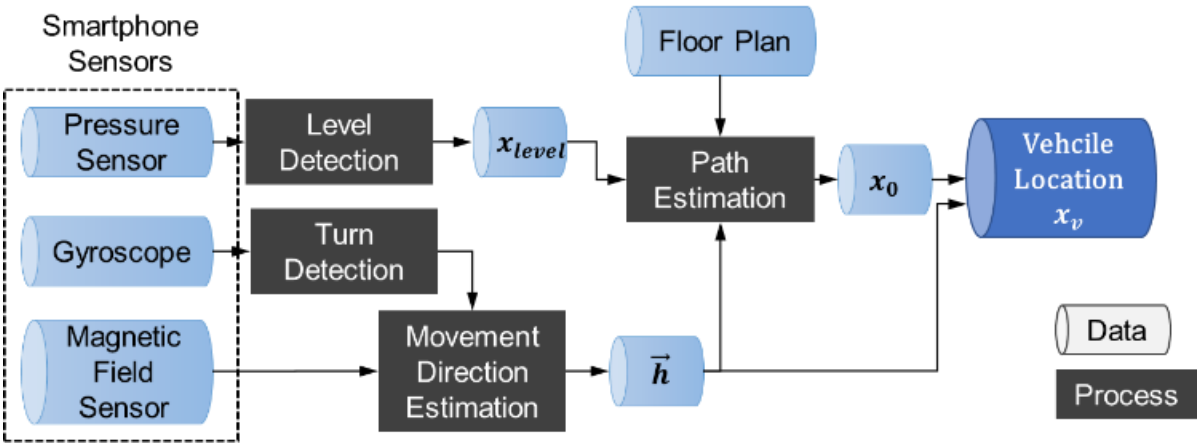


Figure 25: Data and process for Vehicle Dead Reckoning Mechanism.

(1) Turn detection: This module identifies a vehicle's turns by using the data obtained from a gyroscope within a smartphone. First, similar to step detection in the pedestrian dead reckoning mechanism, a low-pass filter is used to eliminate the noise in vertical gyroscope data, and then the numerical integration is applied to obtain the angular displacement. However, due to the noise produced during gyroscope readings in a vehicle, the angular displacement cannot be directly used to determine movement direction. To make the angular displacement measurements more reliable and to determine the type of a turn in a vehicle motion trajectory, the local maxima and the minima are calculated within a moving window, and then compared to distinguish the turns.

(2) Path estimation: This module determines the location of each turn with respect to the building plan. Here, the floor plan of the given building environments is assumed to be known as a priori. The decision tree is constructed based on the all possible sets of turns and level changes in the given floor plan. Then, when a user makes several turns and changes the level in the building, the final location is queried by searching the matching sequence of the turns and level changes in the decision tree. Here, the predetermined turning points at each corner in the given floor plan correct one's paths. For example, in Figure 26, if three right turns are consequently detected after departing from a starting landmark, and the change in the level is detected between the 2nd and 3rd turns, the location of each turn can be estimated as 1, 2, and 3. Here, the paths through the dead reckoning (dot lines) are corrected at each turning location.

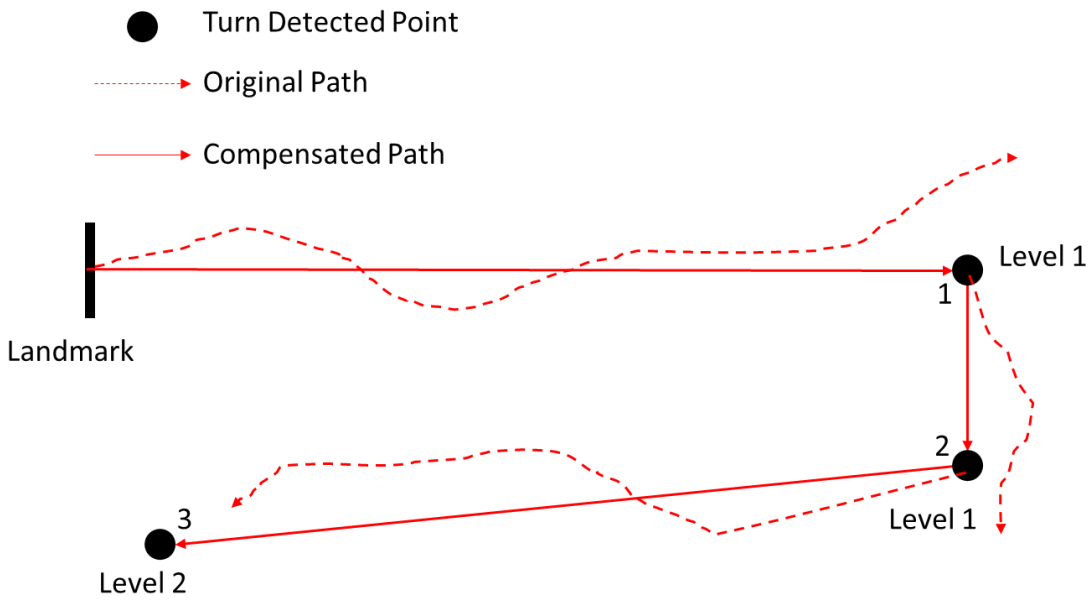


Figure 26: An illustration of the path, turn, and building level estimations using a given building floor plan. When a user makes a turn, the level detection module compares the current pressure sensor reading with the value that was recorded at the previous turn.

4.3 Forward-Backward Event Hazard Localization

Landmark Detection

This module identifies whether a user has reached a certain landmark location, which is done by using machine learning classification. A landmark can be any distinguishable object in a building such as an elevator, a staircase, a door, or fountains. To detect such landmarks from sensory data derived on a commodity smartphone, a Naïve Bayes Classifier is adapted to recognize possible activities of the user as characterized by the landmark. For example, when a user goes up or down a staircase, the vertical acceleration and pressure sensor readings are expected to change on a commodity smartphone. According to Bayes' theorem, the conditional probability of a landmark location (L) with *n* features of the current location can be described by Equation 11:

$$\mathrm{P}(L|F_1, \dots F_i, \dots, F_n) = \frac{\mathrm{P}(F_1, \dots F_i, \dots, F_n|L)\mathrm{P}(L)}{\mathrm{P}(F_1, \dots F_i, \dots, F_n)}, \tag{17}$$

where F_i is the i^{th} feature variable. Depending on the type of landmark, each feature is extracted from the smartphone's sensor readings. Based on conditional probability, the numerator of Equation 11 can be rewritten as the following:

$$\begin{aligned} \mathrm{P}(L)\ \mathrm{P}(F_1, \dots F_i, \dots, F_n|L) &= \mathrm{P}(L)\ \mathrm{P}(F_1|L)\mathrm{P}(F_2, \dots F_i, \dots, F_n|L) \\ &= \mathrm{P}(L)\mathrm{P}(F_1|L)\mathrm{P}(F_2|L, F_1) \dots \mathrm{P}(F_n|L, F_1, F_2 \dots F_{n-1}) \end{aligned} \tag{18}$$

The Naïve Bayes rule follows the assumption that all features are conditionally independent to every other feature. Hence, Equation 12 can be described as follows:

$$\begin{aligned} \mathrm{P}(L|F_1, \dots F_i, \dots, F_n) &= \frac{\mathrm{P}(L, F_1, \dots, F_n)}{\mathrm{P}(F_1, \dots F_i, \dots, F_n)} \\ &= \frac{\mathrm{P}(L)}{\mathrm{P}(F_1, \dots F_i, \dots, F_n)} \mathrm{P}(F_1|L)\mathrm{P}(F_2|L) \dots \mathrm{P}(F_n|L) \\ &= \frac{\mathrm{P}(L)}{\mathrm{P}(F_1, \dots F_i, \dots, F_n)} \prod_{i=1}^{n} \mathrm{P}(F_i|L) \end{aligned} \tag{19}$$

The probability of *i*th feature when a landmark is detected (i.e., $\mathrm{P}(F_i|L)$) is obtained from the mean and standard deviation of the trained data that follow Gaussian distribution. Finally, the landmark is recognized as $\underset{l_j}{\operatorname{argmax}}\, \mathrm{P}\left(L = l_j \middle| F_1, \dots F_i, \dots, F_n\right)$.

Event Detection

In the proposed method, the Forward algorithm first tracks the user's motion trajectory from the starting landmark to the event location, and then the Backward algorithm back-tracks the user's motion trajectory from the post-event landmark. Thus, the timing of the event occurrence needs to be specified to divide the total sensor readings into the two parts for the Forward and Backward approaches respectively. The event occurrence can be determined by classifying into the following two categories:

(1) *Internally detected from a smartphone*: Events such as "making a phone call", "sending a text message", or "taking a picture or video" can be internally detected and recorded in a smartphone.

(2) *Manually triggered by the user before the event occurs*: for those events that can be neither 'directly detected' nor 'automatically classified', the user needs to manually conduct specific activities such as "shake a smartphone" or "press a button" to provide the event time to the proposed localization system.

Forward-Backward Event Localization

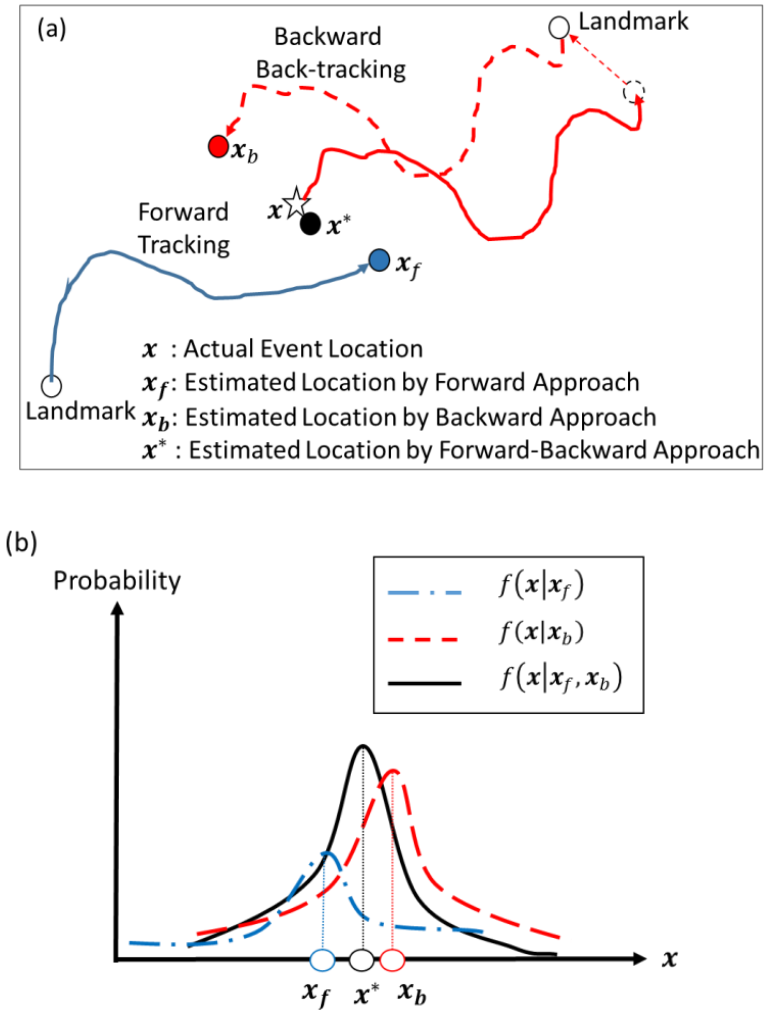


Figure 27: (a): Illustration of the Forward approach, the Backward approach, and the integration, (b): their probability distributions.

Figure 27 illustrates the proposed probabilistic approach for 3D event localization in both indoor and outdoor GPS-denied environments. The proposed solution is infrastructure-freeand leverages accelerometers, gyroscopes, and pressure sensors commonly available in commodity smartphones (Apple and Android devices). This method consists of the following components:

(1) Forward approach: The Forward algorithm tracks a user's location from a *landmark*. Here landmark refers to building features such as entrances, elevators, stairs, and so on, in which their locations are pre-determined and labeled on a building plan. The tracking can be either based on a pedestrian or a vehicle dead reckoning process. When an "event" has occurred, the system estimates the location of the event by using the Forward approach ($\boldsymbol{x_f}$) (Figure 27a).

(2) Backward approach: Similar to the Forward approach, Backward tracking can be done by using either pedestrian or vehicle dead reckoning mechanisms. After an event is observed, the Backward approach tracks the user's position from the actual event location ($\boldsymbol{x}$) to the post-event landmark location as shown in Figure 27a (with a solid line). However, because the actual event location is unknown, the Backward approach estimates the event location ($\boldsymbol{x_b}$) by back-tracking the user's trajectory from the post-event landmark, as shown in Figure27a (with a dashed line).

(3) Probabilistic integration: The forward or backward dead-reckoning process does not affect the result of the other dead-reckoning process, thus the two dead reckoning processes are statistically independent of each other. When two dead reckoning processes are statically independent of each other the integration of the two probability distributions obtained from the Forward and Backward approaches can be described using the following equations:

$$\mathrm{P}\left(\boldsymbol{x}\middle|\boldsymbol{X}_f = \boldsymbol{x}_f, \boldsymbol{X}_b = \boldsymbol{x}_b\right) = \frac{\mathrm{P}\left(\boldsymbol{x}\middle|\boldsymbol{X}_f = \boldsymbol{x}_f\right)\mathrm{P}(\boldsymbol{x}|\boldsymbol{X}_b = \boldsymbol{x}_b)}{\mathrm{P}(\boldsymbol{x})} \propto \mathrm{P}\left(\boldsymbol{x}\middle|\boldsymbol{X}_f = \boldsymbol{x}_f\right)\mathrm{P}(\boldsymbol{x}|\boldsymbol{X}_b = \boldsymbol{x}_b), \quad (20)$$

where x is the actual event location, and $\boldsymbol{X}_f$ and $\boldsymbol{X}_b$ are the random variables of the estimated locations obtained from the Forward and Backward algorithms, and $\boldsymbol{x}_f$and $\boldsymbol{x}_b$ are the realizations (observed values) of each random variable. The conditional probability for localization $\mathrm{P}\left(\boldsymbol{x}\middle|\boldsymbol{x}_f, \boldsymbol{x}_b\right)$ is proportional to the integration of the two probabilities obtained from the Forward algorithm $\mathrm{P}\left(\boldsymbol{x}\middle|\boldsymbol{x}_f\right)$ and the Backward algorithm $\mathrm{P}(\boldsymbol{x}|\boldsymbol{x}_b)$ (Equation 20).

The distribution of the estimated locations obtained by the forward and backward dead-reckoning process would vary depending on the type of sensors and the behavior of pedestrians or vehicles. In this task I have assumed that, in general, the pedestrian and vehicle dead-reckonings follow the Gaussian distribution. If specific distributions of the estimated location obtained from the dead-reckoning process are given, the given PDF can be used without the assumption of a Gaussian distribution.

The two probabilities, $\mathrm{P}\left(\boldsymbol{x}\middle|\boldsymbol{x}_f\right)$ and $\mathrm{P}(\boldsymbol{x}|\boldsymbol{x}_b)$ that follow the Gaussian distribution can be obtained from the Gaussian probability density function (PDF) $f_{\boldsymbol{X}|\boldsymbol{X}_f}\left(\boldsymbol{x}\middle|\boldsymbol{x}_f\right)$ and $f_{\boldsymbol{X}|\boldsymbol{X}_b}(\boldsymbol{x}|\boldsymbol{x}_b)$ in which the mean and standard deviation of the two PDFs are $\boldsymbol{\mu}_{\boldsymbol{X}|\boldsymbol{X}_f} = \boldsymbol{x}_f$, $\boldsymbol{\sigma}_{\boldsymbol{X}|\boldsymbol{X}_f} = \boldsymbol{\sigma}_{\boldsymbol{X}_f}$, $\boldsymbol{\mu}_{\boldsymbol{X}|\boldsymbol{X}_b} = \boldsymbol{x}_b$, and $\boldsymbol{\sigma}_{\boldsymbol{X}|\boldsymbol{X}_b} = \boldsymbol{\sigma}_{\boldsymbol{X}_b}$ respectively. Instead of giving a single best estimate of a user's location obtained from a one-way dead reckoning, the proposed method provides the most likely location ($\boldsymbol{x}^*$) based on the maximum probability of $f\left(\boldsymbol{x}\middle|\boldsymbol{x}_f\right)f(\boldsymbol{x}|\boldsymbol{x}_b)$ (Figure 27 and Equation 21).

$$\boldsymbol{x}^* = \underset{\boldsymbol{x}}{\operatorname{argmax}} f\left(\boldsymbol{x}\middle|\boldsymbol{x_f}, \boldsymbol{x_b}\right) = \underset{\boldsymbol{x}}{\operatorname{argmax}}\left[f\left(\boldsymbol{x}\middle|\boldsymbol{x_f}\right) \times f(\boldsymbol{x}|\boldsymbol{x_b})\right] \tag{21}$$

If any two PDFs follow the Gaussian distribution, it is known that the product of the two PDFs become the scaled Gaussian (Bromiley, 2013). Therefore, the probability distribution of the event location estimated by the integrated method can be expressed as Equation 22. Here, S is the scaling factor of the Gaussian PDF.

$$f_{X|X_f,X_b}\left(\boldsymbol{x}\middle|\boldsymbol{x}_f, \boldsymbol{x}_b\right) = f\left(\boldsymbol{x}\middle|\boldsymbol{x}_f\right) f(\boldsymbol{x}|\boldsymbol{x}_b) = \frac{\boldsymbol{S}}{\sqrt{2\pi}\boldsymbol{\sigma_{fb}}} \exp\left[-\frac{\left(\boldsymbol{x} - \boldsymbol{\mu}_{X|X_f,X_b}\right)^2}{2\boldsymbol{\sigma}^2_{X|X_f,X_b}}\right], \tag{22}$$

$$\text{where } \boldsymbol{\sigma}_{X|X_f,X_b} = \sqrt{\frac{\boldsymbol{\sigma}^2_{X|X_f}\boldsymbol{\sigma}^2_{X|X_b}}{\boldsymbol{\sigma}^2_{X|X_f} + \boldsymbol{\sigma}^2_{X|X_b}}}\ , \qquad \boldsymbol{\mu}_{X|X_f,X_b} = \frac{\left(\boldsymbol{\mu}_{X|X_f}\boldsymbol{\sigma}^2_{X|X_b} + \boldsymbol{\mu}_{X|X_b}\boldsymbol{\sigma}^2_{X|X_f}\right)}{\boldsymbol{\sigma}^2_{X|X_f} + \boldsymbol{\sigma}^2_{X|X_b}},$$

$$and\ \boldsymbol{S} = \frac{1}{\sqrt{2\pi(\boldsymbol{\sigma}^2_{X|X_f} + \boldsymbol{\sigma}^2_{X|X_b})}} \exp\left[-\frac{\left(\boldsymbol{\mu}_{X|X_f} - \boldsymbol{\mu}_{X|X_b}\right)^2}{2(\boldsymbol{\sigma}^2_{X|X_f} + \boldsymbol{\sigma}^2_{X|X_b})}\right]$$

$\boldsymbol{x}^*$ in Equation 2 can be described by Equation 4, where $\boldsymbol{\mu}_{X|X_f}$ and $\boldsymbol{\mu}_{X|X_b}$ are the means of $f\left(\boldsymbol{x}\middle|\boldsymbol{x}_f\right)$ and $f(\boldsymbol{x}|\boldsymbol{x}_b)$ respectively:

$$\begin{aligned}\boldsymbol{x}^* = \underset{\boldsymbol{x}}{\operatorname{argmax}} f\left(\boldsymbol{x}\middle|\boldsymbol{x}_f, \boldsymbol{x}_b\right) = \boldsymbol{\mu}_{X|X_f,X_b} &= \frac{\left(\boldsymbol{\mu}_{X|X_f}\boldsymbol{\sigma}^2_{X|X_b} + \boldsymbol{\mu}_{X|X_b}\boldsymbol{\sigma}^2_{X|X_f}\right)}{\boldsymbol{\sigma}^2_{X|X_f} + \boldsymbol{\sigma}^2_{X|X_b}} \\ &= \frac{\left(\boldsymbol{x}_f\boldsymbol{\sigma}^2_{X_b} + \boldsymbol{x}_b\boldsymbol{\sigma}^2_{X_f}\right)}{\boldsymbol{\sigma}^2_{X_f} + \boldsymbol{\sigma}^2_{X_b}}\end{aligned} \tag{23}$$

The performance of the proposed method can be examined by comparing the expected error from each location. The localization error is the deviation between the actual and the estimated event location, which is calculated using Equation 24. Here, the expected error of $\boldsymbol{X}_f$, $\boldsymbol{X}_b$, and $\boldsymbol{X}^*$ can be described using Equations 25, 26, and 27, respectively.

$$error(\boldsymbol{X}) = \sqrt{(\boldsymbol{X} - \boldsymbol{x}_{event})^2} \tag{24}$$

$$E\big[error(\boldsymbol{X}_f)\big] = E\left[\sqrt{\big(\boldsymbol{X}_f - \boldsymbol{x}_{event}\big)^2}\right] = E\left[\sqrt{\big(\boldsymbol{X}_f - \boldsymbol{\mu}_{\mathbf{X}_\mathrm{f}}\big)^2}\right] = \boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}} \tag{25}$$

$$E[error(\boldsymbol{X}_b)] = E\left[\sqrt{(\boldsymbol{X}_b - \boldsymbol{x}_{event})^2}\right] = E\left[\sqrt{\big(\boldsymbol{X}_b - \boldsymbol{\mu}_{\mathbf{X}_\mathrm{b}}\big)^2}\right] = \boldsymbol{\sigma}_{\mathbf{X}_\mathrm{b}} \tag{26}$$

$$E[error(\boldsymbol{X}^*)] = E\left[\sqrt{(\boldsymbol{X}^* - \boldsymbol{x}_{event})^2}\right] = \boldsymbol{\sigma}_{\boldsymbol{X}|\boldsymbol{X}_f,\boldsymbol{X}_b} = \sqrt{\frac{\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}}^2\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{b}}^2}{\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}}^2 + \boldsymbol{\sigma}_{\mathbf{X}_\mathrm{b}}^2}}, \tag{27}$$

where $\boldsymbol{\mu}_{\boldsymbol{X}_f}$ is the expectation of $\boldsymbol{X}_f$, and $\boldsymbol{\mu}_{\boldsymbol{X}_b}$ is the expectation of $\boldsymbol{X}_b$. Finally, as proved in Equation 28 and 29, the expected error of $\boldsymbol{X}^*$ is smaller than that of $\boldsymbol{X}_f$ and $\boldsymbol{X}_b$.

$$\frac{E[error(\boldsymbol{X}^*)]}{E\big[error\big(\boldsymbol{X}_f\big)\big]} = \sqrt{\frac{\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}}^2\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{b}}^2}{\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}}^2 + \boldsymbol{\sigma}_{\mathbf{X}_\mathrm{b}}^2}}\frac{1}{\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}}} = \sqrt{\frac{\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{b}}^2}{\boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}}^2 + \boldsymbol{\sigma}_{\mathbf{X}_\mathrm{b}}^2}} \leq 1 \ \text{(equal when } \boldsymbol{\sigma}_{\mathbf{X}_\mathrm{f}}^2 = 0\text{)} \tag{28}$$

$$\frac{E[error(\boldsymbol{X}^*)]}{E[error(\boldsymbol{X}_b)]} = \sqrt{\frac{\boldsymbol{\sigma}_{\mathbf{X}_f}^2 \boldsymbol{\sigma}_{\mathbf{X}_b}^2}{\boldsymbol{\sigma}_{\mathbf{X}_f}^2 + \boldsymbol{\sigma}_{\mathbf{X}_b}^2}} \frac{1}{\boldsymbol{\sigma}_{\mathbf{X}_b}} = \sqrt{\frac{\boldsymbol{\sigma}_{\mathbf{X}_f}^2}{\boldsymbol{\sigma}_{\mathbf{X}_f}^2 + \boldsymbol{\sigma}_{\mathbf{X}_b}^2}} \leq 1 \ \ (\text{equal when } \boldsymbol{\sigma}_{\mathbf{X}_b}^2 = 0\,) \qquad (29)$$

4.4 Validation Test

The proposed infrastructure-free localization method using a smartphone was validated through several case studies. These experiments were conducted in the context of localizing the parking location of a vehicle in a multistory parking garage, a process that requires both vehicle and pedestrian dead reckoning for the event localization.

Figure 28 illustrates an overview of the data and a process for the case study. When the vehicle reaches the starting landmark (e.g., speed bumps at the entrance), the first phase of the system begins to track the vehicle until it reaches the parking location based on the vehicle's dead reckoning. Here, 'parking' is considered an event. After the event occurs, the second phase of the system back-tracks the driver from the final landmark location (e.g., elevator or stairway in the building) by using pedestrian dead reckoning.

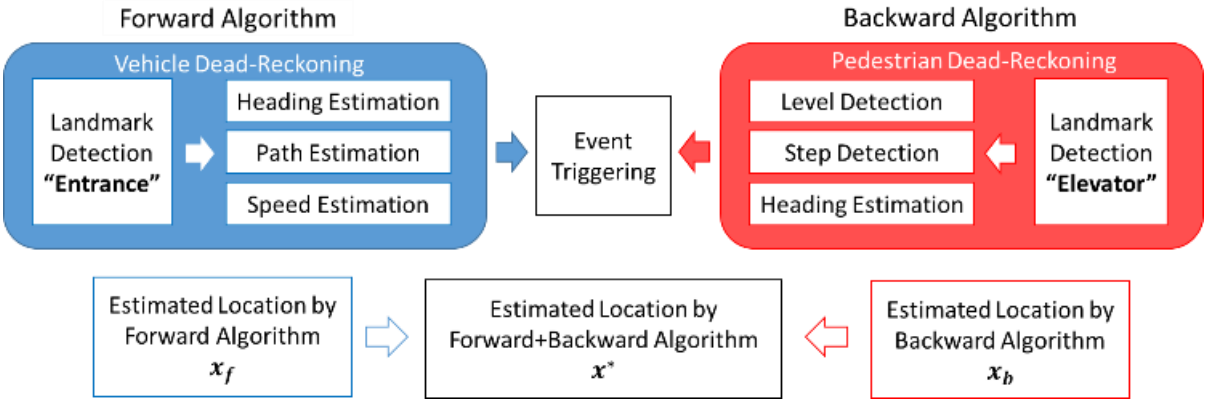


Figure 28: Probabilistic localization of parking places in a multistory parking garage.

Experimental Setups

The experiments were conducted at a parking garage located on campus of the University of Illinois at Urbana-Champaign (Figure 29a). At the time of the study, the parking garage had a total of 1,495 parking spaces on six at and included a roof, along with 21,300 square foot of retail space at ground level. Several case studies on the performance of the event localization method were conducted, and their locations are shown in Figure 29b.

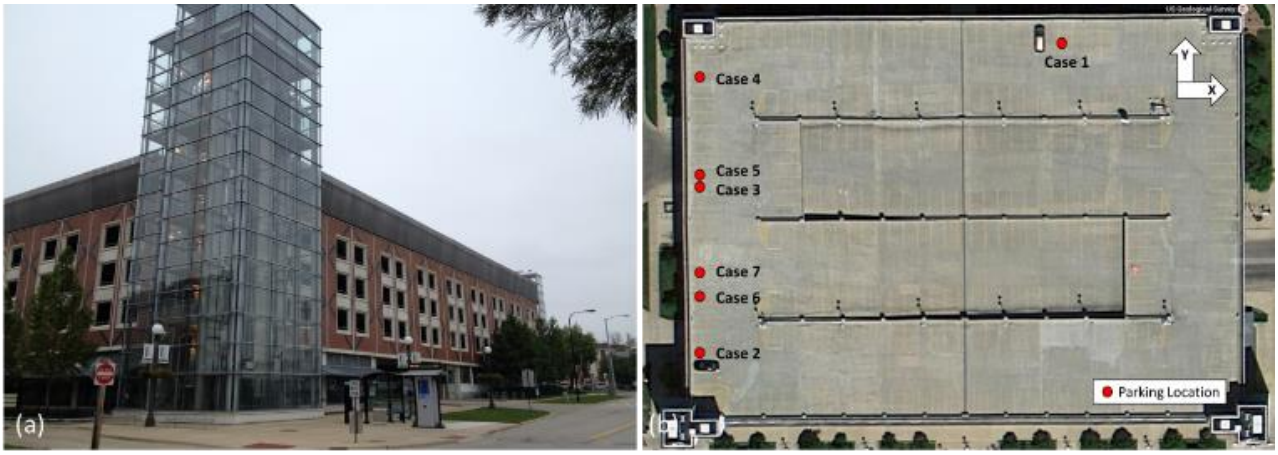


Figure 29: (a): the parking garage, and (b): experimental setups.

The event locations of cases #1-4 were on the roof top (6F), and those of cases #5-7 were on 5F. A Samsung Galaxy S4 smartphone which has a 3-axis gyroscope, a 3-axis accelerometer, a 3-axis magnetic-field sensor, and a pressure sensor was used to track the motion trajectories of the vehicle and the mobile user. To collect and record the embedded sensor data from the smartphone, an android-based sensor data collection application was developed. In the experiments, all sensor readings were recorded at a sampling rate of 10 Hz.

Forward Approach: Vehicle Tracking

In the experiments, the Forward algorithm tracks the vehicle from a starting landmark location to the parking location by using 3D automotive dead reckoning. In this case study, a speed bump at the entrance of the parking building is considered a the starting landmark. Once the starting landmark is detected, the Forward algorithm tracks the vehicle until the parking event is detected. Then, the probability distribution on the vehicle location ($\boldsymbol{x_f}$) is estimated as described in Section 3.2.2. For detecting turns, the vertical gyroscope data was first obtained (Figure 30a) and then pre-processed with the low-pass filter. On the basis of the difference between the negative and positive picks of the numerical integration of the vertical gyroscope data (Figure 30b), the vehicle turns are classified into right, left, or U-turn directions. Figure 30c shows the detected turns along with the movement directions of the vehicle.

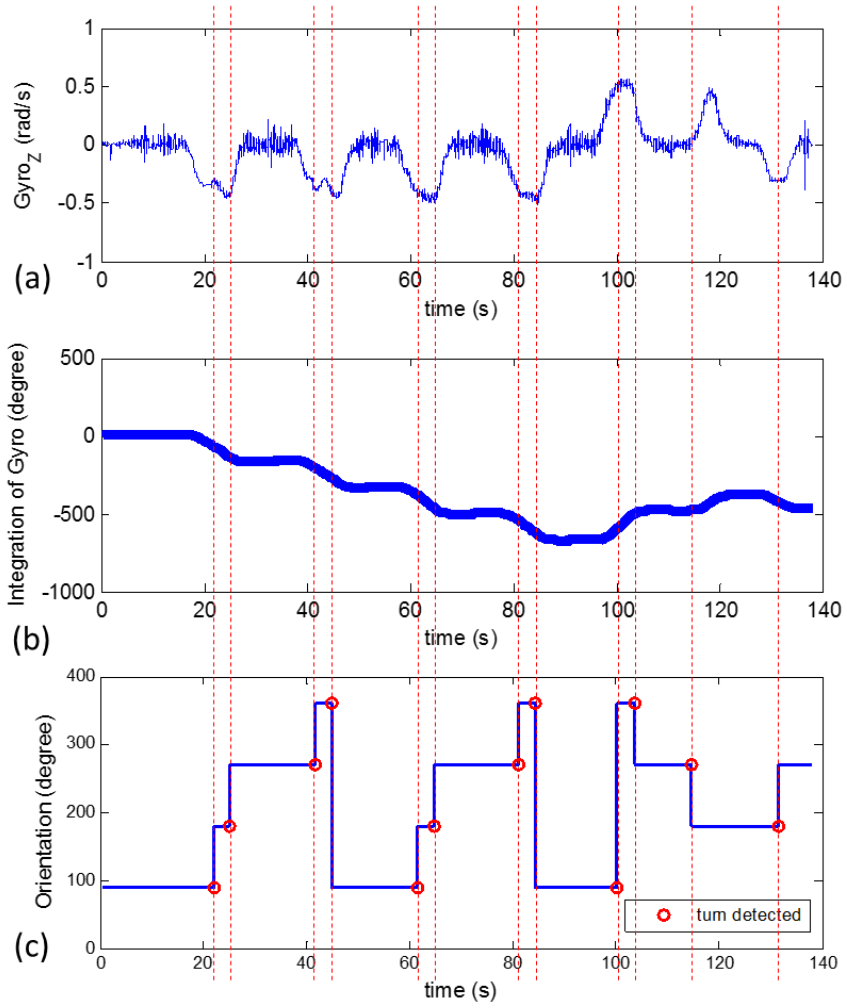


Figure 30: (a): vertical gyroscope data, (b): the numerical integration of the gyroscope data, (c): the movement direction of the vehicle and its detected turns.

Figure 31 shows the identified building level where the vehicle is located. As shown in this figure, based on ΔP at two consecutive turning points, five changes in building levels are detected as the vehicle moves towards the upper levels. The threshold value vary depending on the type of buildings. Here, according to a floor-to-floor height of about 3.5m in the multistory parking garage, the value is assigned as 1.5hPa. In Figure 31, the turning locations are marked as circles to represent the pressure sensor data.

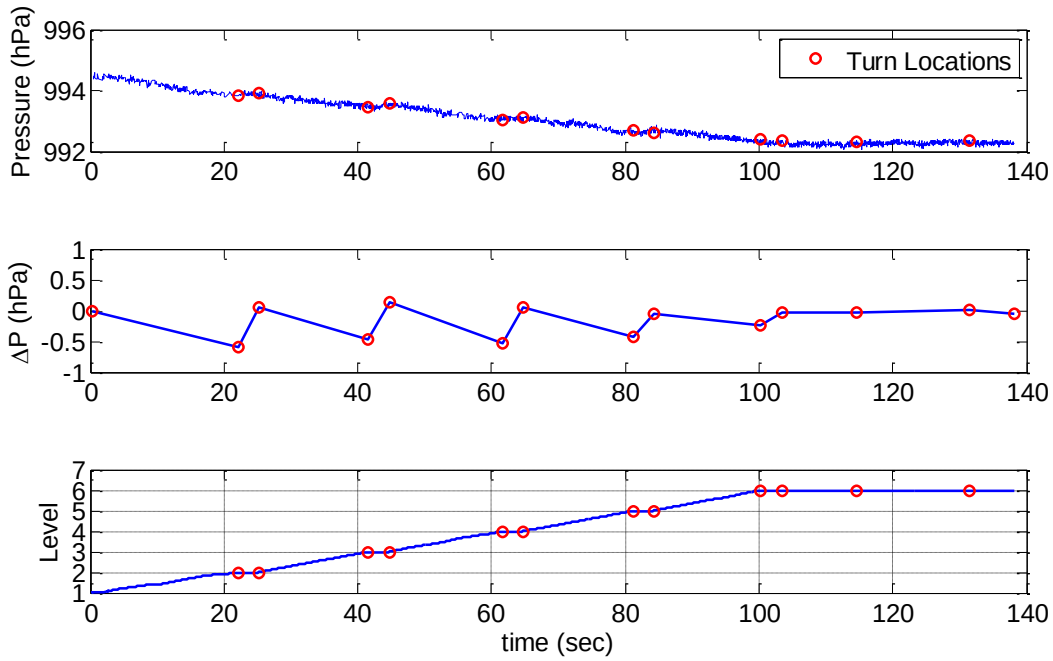


Figure 31: Level detection of the vehicle using pressure sensor data.

In the instance of moving vehicles, directly finding distinct and consistent peak points – analogous to human footsteps–from the acceleration data is challenging. In the conducted experiments, the vehicle speed was estimated based on (1) each distance between two consecutive turn locations identified in the given floor plan and (2) the driving time taken for passing each section. To do that, the vehicle speed was modeled as a trapezoid shape (Figure 32). In this model, the vehicle accelerated at the beginning ($T0-T1$) or after stops ($T4-T5$), decelerates before stops ($T2-T3, T6-T7$), and the vehicle had a speed of V_0 between acceleration and deceleration ($T1-T2, T5-T6$).

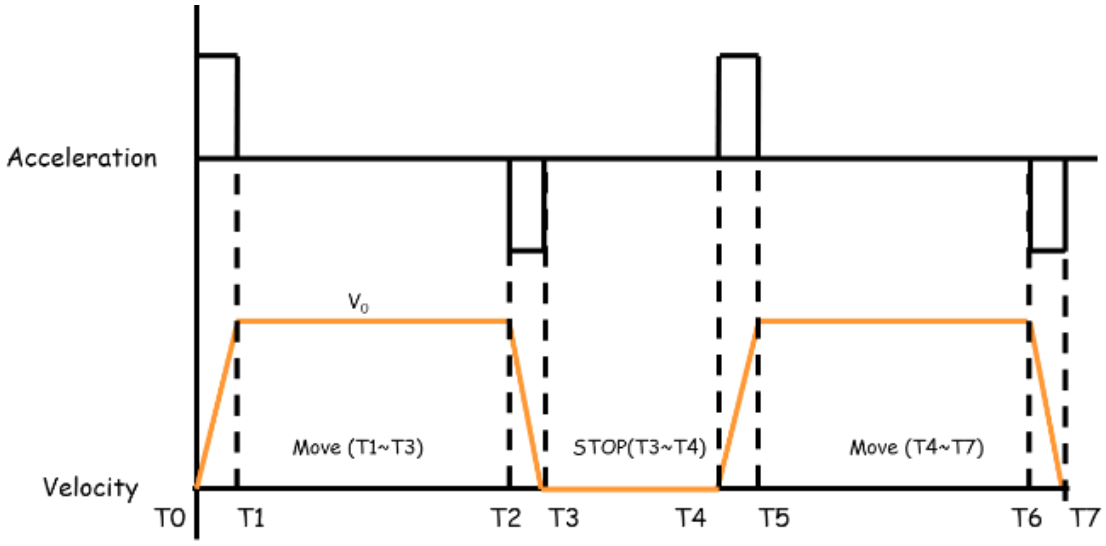


Figure 32: Vehicle speed model in the form of a trapezoid.

The presence of other vehicles or pedestrians in the parking building can require the user in the moving vehicle to make frequent stops. To consider this, the stop detection module determines whether or not the vehicle is moving by using the vertical acceleration data obtained from the smartphone. Depending on the roughness of the pavement, moving vehicles typically have some degree of variation in vertical acceleration, and thus the vertical movement of the vehicle will be close to zero when the vehicle stops. By detecting the time periods that maintain the constant vertical acceleration, the stop time of the vehicle is calculated, and then is deduced from the total driving time taken for passing each section. The number of stops is notated as n_{stop}, and the total driving time between two consecutive turning points (i.e., a section) is calculated by subtracting the stop time (t_{stop}) from the total elapsed time (t_0). Then, based on the trapezoid vehicle speed model (Figure 32), the distance between two consecutive turns (d) can be described as Equation 30:

$$d = 2n_{stop}\left(\frac{1}{2}V_0\tau\right) + V_0\left(t_0 - t_{stop} - 2n_{stop}\tau\right) = V_0\left(t_0 - t_{stop} - n_{stop}\tau\right), \quad (30)$$

where τ is the time required for (1) stopped vehicle to accelerate to reach the speed (V_0) and (2) moving vehicle to decelerate to the stop. This value varies depending on the specification of vehicles and the pavement conditions. In this case study, this value was selected as 2.5 seconds according to several experiments on the driver. For *i* th section, the speed V_0 is calculated by using the following equation:

$$V_{0,i} = \frac{\mathrm{d}}{t_0 - t_{stop} - n_{stop}\tau} \quad (31)$$

According to the variation of the vehicle speed, V_0 is assumed to follow the normal distribution; i.e., ($V_0 \sim N\left(\mu_{v_0}, \sigma_{v_0}\right)$. Here, the mean ($\mu_{v_0}$) and the standard deviation (σ_{v_0}) of V_0 are calculated by using the following equations with the number of sections (n):

$$\mu_{v_0} = \frac{1}{n}\sum_{i}^{n} V_{0,i} \quad (32)$$

$$\sigma_{v_0}^2 = \frac{1}{n}\sum_{i}^{n}\left(V_{0,i} - \mu_{v_0}\right)^2 \quad (33)$$

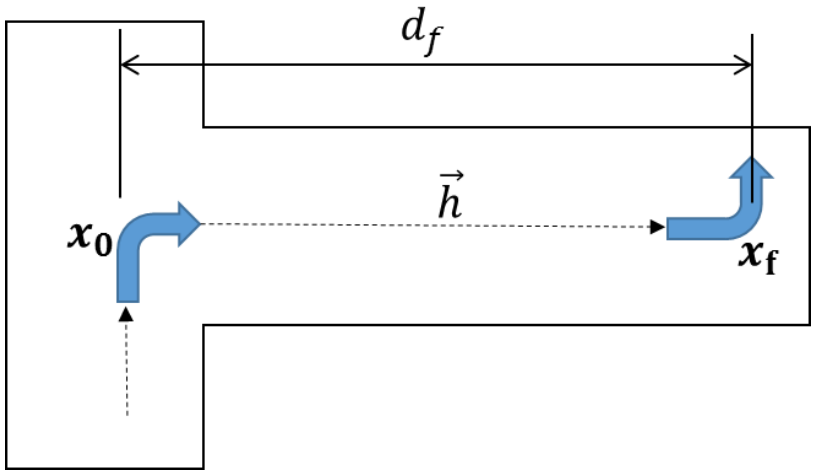


Figure 33: Final location (x_f) obtained from the Forward algorithm.

As shown in Equation 34, the final location obtained from the Forward algorithm ($\boldsymbol{x_f}$) can be described by using the last turn location ($\boldsymbol{x_0}$), the heading direction ($\vec{\boldsymbol{h}}$), and the distance that the vehicle moved at the last section (d_f) (from $\boldsymbol{x_0}$ to $\boldsymbol{x_f}$ in Figure 33). Here, $\boldsymbol{x_0}$ and $\vec{h}$ are obtained from the path estimation module, and d_f is calculated based on the vehicle speed (v_t) and the time taken for moving from $\boldsymbol{x_0}$ to $\boldsymbol{x_f}$.

$$\boldsymbol{X_f} = \boldsymbol{x_0} + \vec{\boldsymbol{h}} d_f = \boldsymbol{x_0} + \sum_t \vec{\boldsymbol{h}}(v_t \Delta t) = \boldsymbol{x_0} + \vec{\boldsymbol{h}} \cdot V_0 \cdot t_0 \tag{34}$$

Here, t_0 is $(t - n_{stop}\tau - t_{stop})$. Because V_0 is characterized as the normal distribution, $\boldsymbol{X_f}$ will also follow a normal distribution; i.e., $\boldsymbol{X_f} \sim N(\boldsymbol{\mu_{X_f}}, \sigma_{X_f})$ with the mean ($\boldsymbol{\mu_{X_f}}$) and the standard deviation (σ_{X_f}) which can be calculated by using the following equations:

$$\boldsymbol{\mu_{X_f}} = E[\boldsymbol{X_f}] = E[\boldsymbol{x_0} + \vec{h} \cdot V_0 \cdot t_0] = \boldsymbol{x_0} + \vec{h} \cdot \mu_{v_0} \cdot t_0 \quad (35)$$

$$\sigma^2_{X_f} = Var[\boldsymbol{x_f}] = Var[\boldsymbol{x_0} + \vec{h} \cdot V_0 \cdot t_0] = \sigma^2_{v_0} \cdot t_0 \quad (36)$$

Backward Approach: Driver Tracking

Once the vehicle is parked, the Backward algorithm tracks the driver who is likely to approach one of the nearby landmarks in the given floor plan—such as an elevator or a staircase. To do that, by using multi-metric sensor data obtained from the smartphone equipped with the driver, the probability distribution of the driver location ($\boldsymbol{x_b}$) is estimated through the pedestrian dead reckoning process described in the previous section.

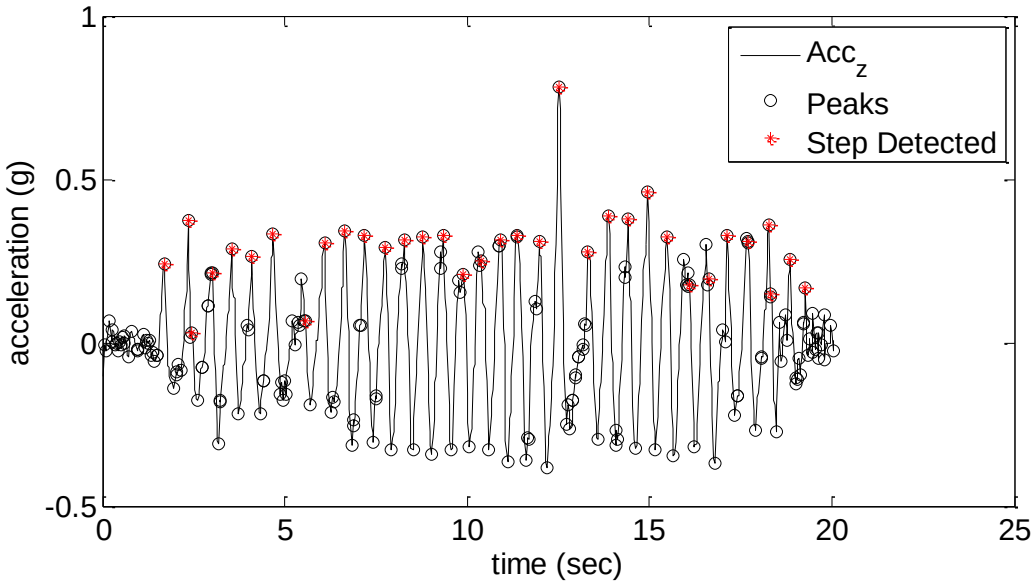


Figure 34: Step detection using the vertical accelerometer data obtained from a smartphone.

The Backward approach first tracks the driver from the event location to the landmark through step detection (Figure 34) and movement direction estimation modules. In this case study,

the threshold for step detection is set as 0.2 of the maximum peak, a threshold based on several experiments. Then, by recognizing the driver's activities relevant to the specific landmark through the use of the machine learning classification, the end-point of the driver trajectory is detected. Once the driver reaches the landmark, the Backward algorithm back-tracks the user's motion trajectory from the detected landmark.

In the conducted experiments, an elevator was selected as a post-event landmark. An elevator can be detected by using the pressure sensor readings obtained from a smartphone. Typically, the pressure data on a commodity smartphone varies based on different altitudes. Here, the changes in the pressure sensor readings are used to determine the movement of the elevator. However, the pressure sensor readings typically have temporal variations (as shown in Table 5), which impede the direct application of the pressure sensor readings to estimate the user's altitude. To overcome this challenge, the ascending or descending rates of the pressure sensor readings that are typically consistent over time are used to determine whether the elevator is going up, down, or is stopped. Here, the differences in pressure sensor readings are used as a feature for classifying the movements of the elevator.

Table 5: Pressure sensor readings for different days (hPa).

	Day 1	*Day 2*	*Day 3*	*Day 4*	*Day 5*
1st Floor	986.0	987.7	1006.2	994.3	993.3
5th Floor	984.0	989.7	1004.3	992.5	991.5
Difference	-2.0	-2.0	-1.9	-1.8	-1.8
Diff/Floor	-0.5	-0.5	-0.475	-0.45	-0.45

Figure 35 (a) and (b) show 'going up' from the 1st floor to the 5th floor while stopping once at the 4th floor and 'going down' from the 5th floor to the 1st floor while stopping once at the 4th floor, which are movements consistent with actual elevator movements occurring at the time of data collection.

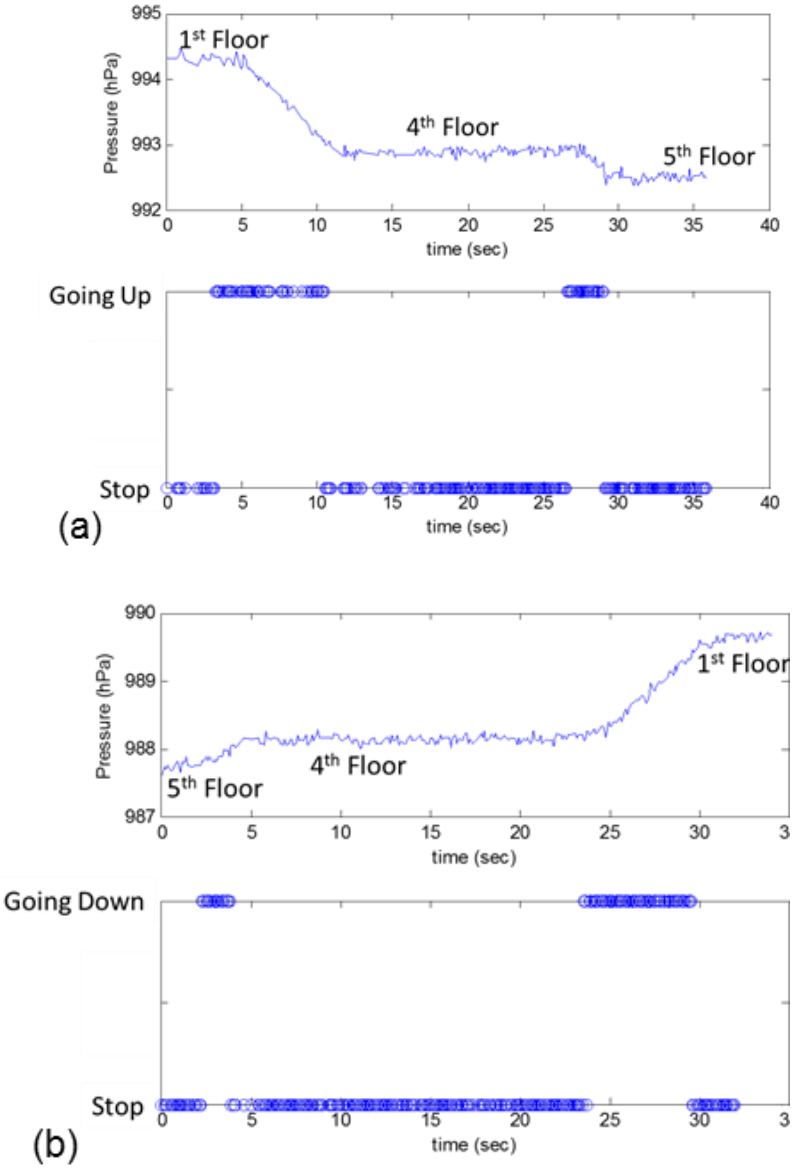


Figure 35: Landmark detection (elevator). (a): going up, and (b): going down

Finally, the back-tracked location ($\boldsymbol{X_b}$) is estimated by subtracting the trace of the driver from the landmark location (Equation 37 and Figure 36), which is described by using the detected landmark location ($\boldsymbol{x_{LM}}$) in the given floor plan, the heading direction ($\vec{\boldsymbol{h}}$) obtained from the movement direction estimation module, the stride length (L_s), and the number of steps ($\boldsymbol{n_{step}}$).

$$\boldsymbol{X_b} = \boldsymbol{x_{LM}} - \sum_{i=1}^{\boldsymbol{n_{step}}} \vec{\boldsymbol{h}}_{\boldsymbol{i}}(L_s) \tag{37}$$

Here, $\boldsymbol{X_b}$ is assumed to follow the normal distribution (i.e., $\boldsymbol{X_b} \sim N(\boldsymbol{\mu_{X_b}}, \sigma_{X_b})$). The mean and standard deviation of L_s were set as 0.8m, and 0.04, which were experimentally validated in (Collins and Kuo 2013). The mean and the standard deviation of $\boldsymbol{X_b}$ can be calculated by using the following equations.

$$\boldsymbol{\mu_{X_b}} = \mathrm{E}[\boldsymbol{X_b}] = \mathrm{E}\left[\boldsymbol{x_{LM}} - \sum_{i=1}^{\boldsymbol{n_{step}}} \vec{\boldsymbol{h}}_{\boldsymbol{i}}(L_s)\right] = \boldsymbol{x_{LM}} - \mathrm{E}\left[\sum_{i=1}^{\boldsymbol{n_{step}}} \vec{\boldsymbol{h}}_{\boldsymbol{i}}(L_s)\right] \tag{38}$$

$$\boldsymbol{\sigma^2_{X_b}} = \mathrm{Var}[\boldsymbol{X_b}] = \mathrm{Var}\left[\boldsymbol{x_{LM}} - \sum_{i=1}^{\boldsymbol{n_{step}}} \vec{\boldsymbol{h}}_{\boldsymbol{i}}(L_s)\right] = \sigma^2_{L_s}(\boldsymbol{n_{step}}) \tag{39}$$

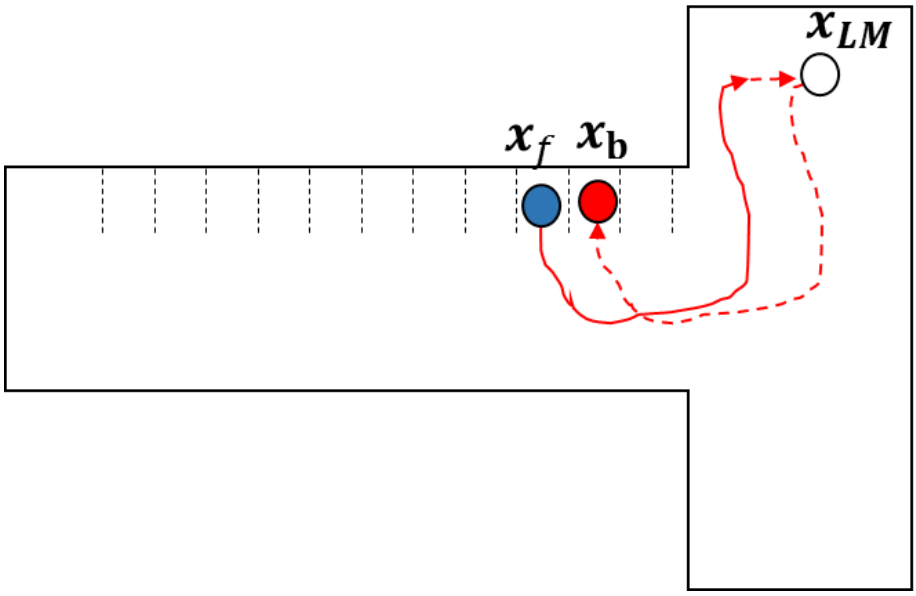


Figure 36: Final location (x_b) obtained from the Backward algorithm.

Integration of the Forward and Backward Localization Probability Distributions

The final step for localizing the event is to integrate the probability distributions obtained from the Forward and Backward approaches. Because 'the parking location by tracking the vehicle from the entrance' and 'the parking location by back-tracking the driver from the landmark' are statistically independent from each other, the most probable parking location can be determined by choosing the location that has the maximum probability as established by the use ofthe integrated probability distribution (Equation 21).

Experimental Results

Table 6 and Figure 37 illustrate the experimental results on the event localization using (1) the Forward algorithm, (2) the Backward algorithm, and (3) the proposed integrated method, respectively. Table 2 presents the actual event location and the localization results with x, y

coordinates, and the error ($meters$), plus the z coordinates (floors). The origin of the coordinate system is the entrance of the parking building located on the first floor. Figures 38 and 39 illustrate the conditional PDF obtained by Equation 3 for the direction that is perpendicular to the parking direction (i.e., x-axis for case #1, and y-axis for case #2-7, as shown in Figure 29b).

Table 6: Experimental results on seven case studies.

Case	Actual			Forward				Backward				Forward + Backward			
	x(m)	y(m)	z	x(m)	y(m)	z	Error	x(m)	y(m)	z	Error	x(m)	y(m)	z	Error
1	75.0	26.0	6	74.7	32.0	6	6.0	73.5	20.0	6	6.2	73.1	24.6	6	2.4
2	3.0	-36.0	6	0.4	-51.5	6	15.7	7.0	-40.2	6	5.8	2.7	-40.4	6	4.4
3	3.0	-3.0	6	1.5	-15.0	6	12.1	6.3	6.0	6	9.6	1.9	2.8	6	5.9
4	3.0	18.0	6	3.1	18.6	6	0.6	9.1	26.3	6	10.3	3.3	18.6	6	0.7
5	3.0	-1.0	5	0.3	-21.8	5	21.0	7.0	10.2	5	11.9	0.6	-0.9	5	2.4
6	3.0	-25.0	5	-1.4	-49.2	5	24.6	6.3	-21.3	5	5.0	0.0	-22.0	5	4.2
7	3.0	-20.0	5	0.5	-37.5	5	17.7	2.8	-19.9	5	0.2	0.8	-20.6	5	2.3
Avg.	-	-	-	-	-	-	13.95	-	-		6.99	-	-		3.18

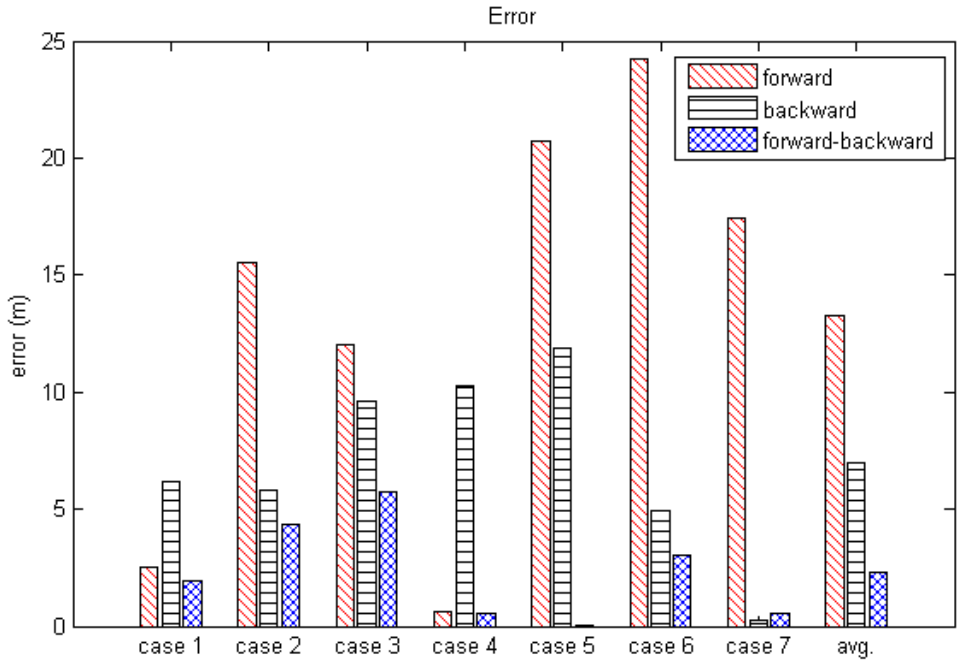


Figure 37: Localization errors using the Forward algorithm, the Backward algorithm, and the proposed method in seven case studies.

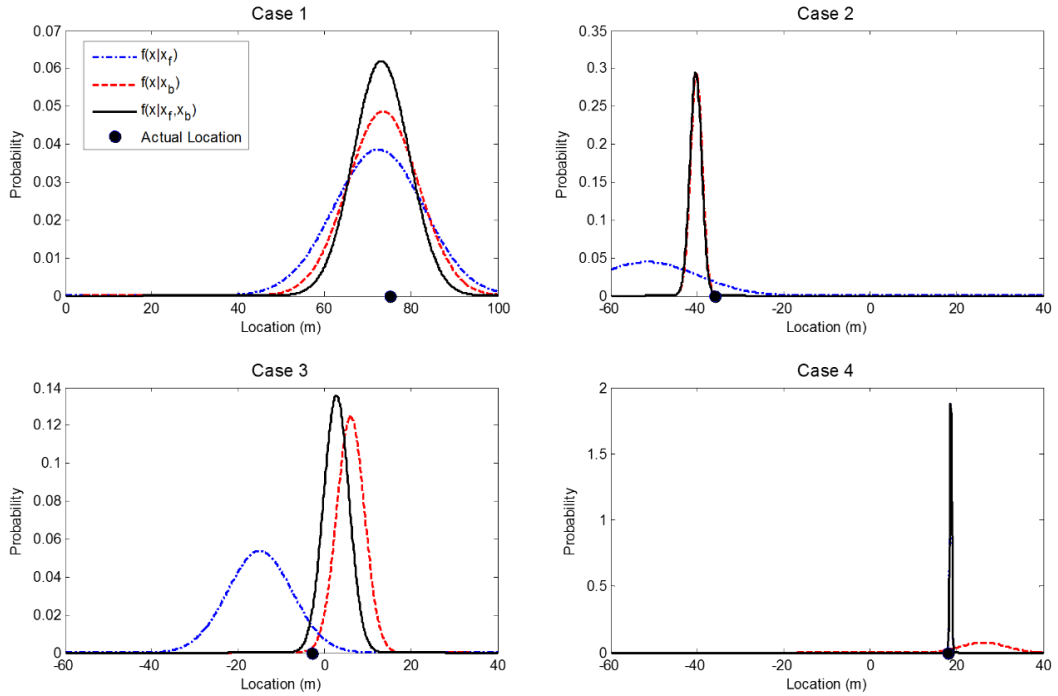


Figure 38: Conditional PDF of Cases #1~#4, using the Forward algorithm, the Backward algorithm, and the proposed Forward-Backward method.

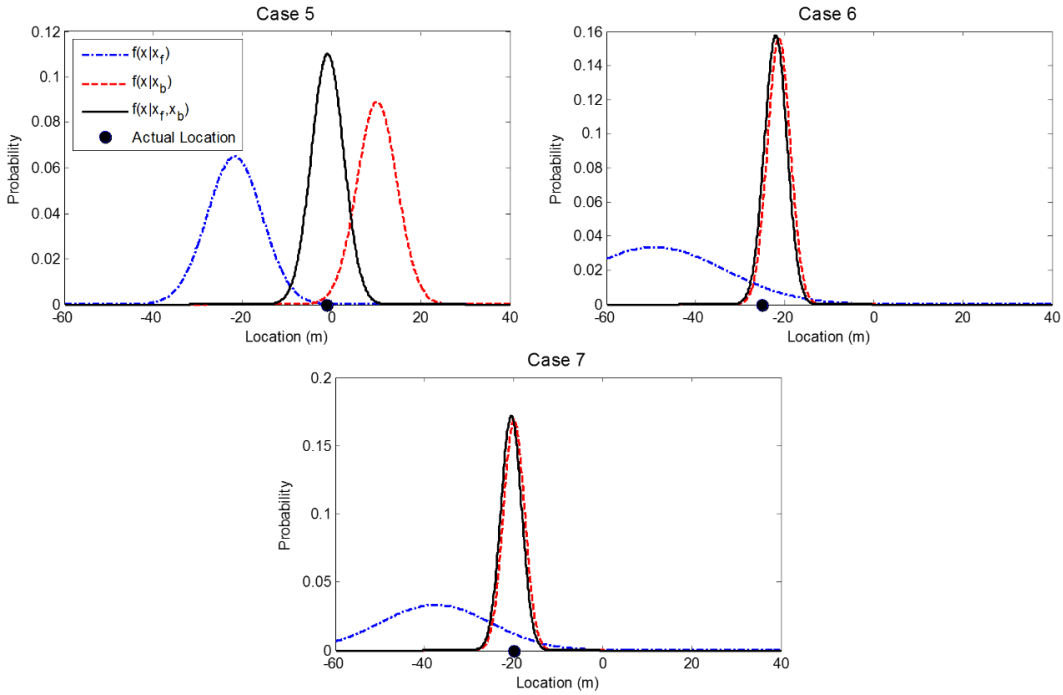


Figure 39: Conditional PDF of Cases #5~#7, using the Forward algorithm, the Backward algorithm, and the proposed Forward-Backward method.

As shown in Table 6, the average localization error on the Forward approach was 13.95m. As can be seen in the PDF (Figures 38 and 39), the range of the event location estimated by the Forward approach is affected by the variance ($\boldsymbol{\sigma}_{\boldsymbol{X}_f}^2$), which is attributed to the following two variables (Equation 36): (1) the variance of the vehicle speed estimation ($\sigma_{v_0}^2$), which exists because of different driving behaviors in a parking garage; and (2) the driving time from the last turning point to the event location (t_0). When this time gap is longer, the event localization by the Forward approach has a higher probability of producing a larger error: i.e., the expectation on the localization error becomes higher. For example, in cases where the location of an event is physically farther from the last turning point (e.g., Case #6), the Forward approach is expected to produce a larger error. Similarly, in cases where the location of the event is closer to the last turning point (e.g. Case #4), a lower error in event localization is expected.

The average localization error on the Backward approach was 6.99m. Similar to the Forward approach, the localization results from the Backward approach also have some degree of variation ($\boldsymbol{\sigma}_{\boldsymbol{X}_b}^2$) (Figures 38 and 39), which is attributed to the following two variables (Equation 39): (1) the variance of the stride length ($\sigma_{L_s}^2$) and (2) the number of steps ($\boldsymbol{n}_{step}$): This value is proportional to the distance between the event location and the landmark location. Here, when $\boldsymbol{n}_{step}$ has higher values, the event localization determined by the Backward approach has a higher probability of producing a larger error--i.e., the expectation that the localization error becomes higher. For example, when the distance between the parking location and the landmark is long (e.g., Case #4), the drift errors accumulated in proportion to the user's walking distance will result in a higher margin of error with regard to localization.

Finally, the average localization error using the proposed integrated Forward-Backward method was 3.18*m*. The solid lines in Figures 38 and 39 illustrate the integration of two separate Gaussian PDFs, obtained from the Forward and Backward algorithms. A one-way localization using either the Forward or Backward approach is typically underestimated or overestimated due to accumulated drift errors. Here, all possible cases for integration have been classified into the following two cases, with equal probability:

(1) Both the Forward and Backward approaches either underestimate or overestimate the event location: In certain cases, (e.g., Case #1-6), the two different locations estimated by the Forward and Backward approaches will be positioned on the opposite side with respect to the actual event location. Thus, by integrating two probability distributions, the drift errors on event localization are most likely to be reduced; and

(2) Either the Forward or Backward approach overestimates the location, and the remaining approach underestimates the event location: In this case (e.g., Case #7), because two estimated locations using the Forward and Backward approaches will be positioned on the same side with respect to the actual event location, one of the two one-way approaches (e.g., the Backward approach in Case #7) would yield lower localization errors compared to the integrated method. Nonetheless, such cases occur with an equal probability for both the Forward and Backward approaches. Also, even when the one-way dead reckoning approach produces a lower error compared to the integrated method, the estimated location using the integrated method becomes very close to that of the best one-way dead reckoning with high probability. Thus, in terms of the expectation of localization errors, the integrated method still produces a lower expectation of localization errors than any one-way dead reckoning in this case (as demonstrated in Equation 20 and 21).

Ultimately, comparing the accuracy of one-way localization, which is highly influenced by the distance from the reference points (e.g., the last turning point of the Forward approach and the landmark of the Backward approach) to the event location, the proposed integrated method can localize an event more accurately.

4.5 Summary

This chapter has proposed an infrastructure-free approach for 3D event localization on commodity smartphones. The method is built on three key modules: (1) landmark detection, (2) 3D dead reckoning for pedestrian and vehicle, and (3) event detection. This method reliably operates in both indoor and outdoor GPS-denied environments and does not require any dedicated infrastructure. Unlike localization technologies that depend on external infrastructure, such as GPS, WLAN, and RFID, the proposed method leverages existing multi-metric sensors, which are embedded in commodity smartphones. First, the Forward algorithm tracks a user's motion trajectory from a formerly determined position and advances that position until the user reaches an event location. Once the event has occurred, the Backward algorithm calculates the location of the event by back-tracking the user's motion trajectory from a detected landmark, using a machine learning-based classification. Finally, the proposed method integrates the two probability distributions and, based on total maximum probability, derives the most-likely location for an event. To validate the proposed method, several case studies for parking events were conducted in the already-specified multistory parking garage. The experimental results show that integrating multiple dead-reckonings based on different reference points can estimate the most likely location of an event with a reasonable level of accuracy. The proposed 3D event localization method is expected to be utilized for various civil engineering applications, including localizing first

responders at building emergency scenes, localizing building elements in need of improvement for retrofit purposes, and construction field reporting. Future work includes reducing the effects of variation in a smartphone's orientation, reducing the battery consumption of the smartphone for achieving energy efficient implementation, inferring accurate vehicle speeds without the need for additional sensors, and minimizing the computational cost to integrate two PDFs for improving the applicability in civil and infrastructure engineering domains.

CHAPTER 5 STRUCTURAL MONITORING USING COMMERCIAL CAMERAS

In this chapter, contactless measurement configuration for dynamic motion, using computer vision techniques, is suggested for structural system identification with regard to practical applications. In order to overcome the limitations found in previous studies while at the same time making use of the advantages provided by image-based algorithms, three main contributions are proposed here: 1) an automated process of displacement measurements from a given structure and overall framework for structural system identification is suggested, 2) target-free displacement measurements of dynamic motion from a structure in which the target markers is unnecessary to be attached is provided, and 3) suggested use of a customer-level camera and its requirement for the system identification application in terms of frame rate, temporal aliasing and image resolution issues is put forth. The proposed technique will be verified with structure dynamic tests and corresponding system identification, including the earthquake response of the structure. The results are carefully compared with data from a conventional wired system. The suggested method shows great potential for practical application to structures, as well as the use of commercially available cameras. This work reports on a method that provides a new diagnostic capability and a cost-effective displacement measurement configuration. Its use in system identification is one that has not provided by previous research. This work aims to broaden and optimize the application of the computer vision techniques to civil infrastructure, particularly for large-scale infrastructures subjected to dynamic loading.

5.1 Framework Development

For simple and robust system identification for civil infrastructures, this task proposes a framework for target-free, vision-based system identification using commercial cameras (e.g. GoPro, smartphones). Figure 40 shows an overview of the proposed framework. The framework is composed of three main components: (1) Camera calibration, (2) the vision-based displacement measurement, and (3) system identification.

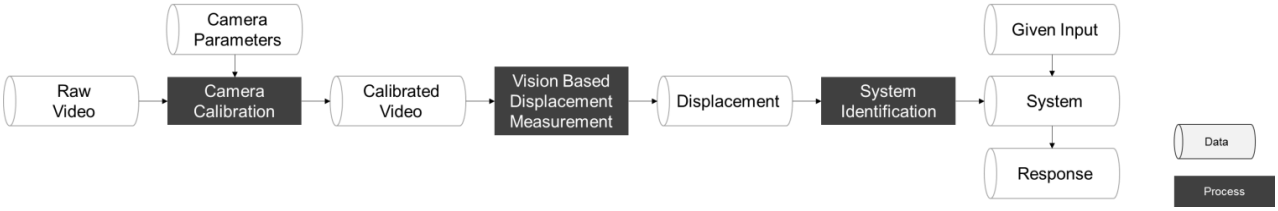


Figure 40: Flowchart for Target-Free Vision Based System Identification Framework.

Camera calibration

Most cameras, especially commercial ones, have optical distortion artifacts that cause a displacement error for a recorded object. Images in every frame of the video the researchers recorded were calibrated and restored into a video. To remove distortion effects, the camera calibration method by (Zhang 2000) was used. By using the camera calibration method, the camera's intrinsic matrix can be estimated. Once the intrinsic matrix is obtained, the matrix can be used to remove distortions for any image frames recorded by the camera.

The Vision Based Displacement Measurement

Once the distortions in the video are removed, the dynamic response of structures can be measured by analyzing the video frame by frame. The proposed system can again be categorized into four components as shown in Figure 41: (1) Region of the Interest (ROI) box selection, (2) feature

detection, (3) feature tracking, and (4) outlier detection. From the calibrated videos, the regions of interest (ROI), which indicate the location of the object to be tracked, are selected. Each ROI is selected by drawing a box on the tracked object in the first frame of the video;', however, some constraints must be considered. These will be discussed in more detail in Section 3.

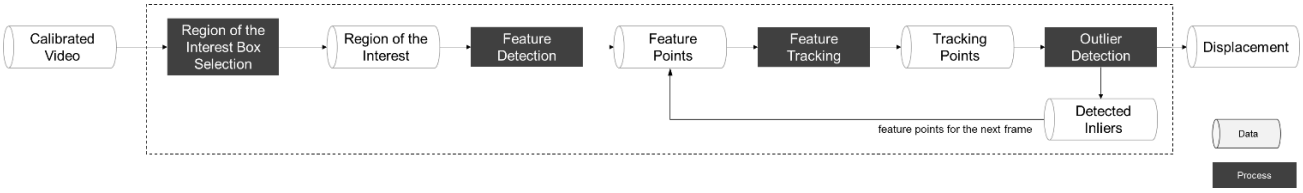


Figure 41: Flowchart for vision based displacement measurement.

In order to achieve a reliable tracking procedure, distinct features need to be detected from the object of interest. There are several feature detection methods that can be used for that purpose. In this research study the corner detection method suggested by (Harris and Stephens, 1988) is utilized to find the features in the initial frame. The Harris corner method is consistent with algorithms that depend on the spatial intensity gradient of the images to find their trajectories, such as the KLT tracker, which will be discussed in the next section. In the Harris corner method, the weighted sum of the squared differences between two image patches with a small shift (x, y) is utilize, which can be expressed as:

$$\mathrm{E}_{\mathrm{x,y}} = \sum_{u,v} w_{u,v}\left[I_{x+u,y+v} - I_{u,v}\right]^2 = \sum_{u,v} w_{u,v}\left(\frac{\partial \mathrm{I}}{\partial \mathrm{x}}x + \frac{\partial \mathrm{I}}{\partial \mathrm{y}}y\right)^2 = (x\ y)\mathbf{M}(x\ y)^T \tag{40}$$

, where $I_{u,v}$ is the original 2-dimensional image and $I_{x+u,y+v}$ is the shifted image, and

$$\mathbf{M} = \begin{bmatrix} \left(\frac{\partial \mathrm{I}}{\partial \mathrm{x}}\right)^{\mathbf{2}} & \left(\frac{\partial \mathrm{I}}{\partial \mathrm{x}}\right)\left(\frac{\partial \mathrm{I}}{\partial \mathrm{y}}\right) \\ \left(\frac{\partial \mathrm{I}}{\partial \mathrm{x}}\right)\left(\frac{\partial \mathrm{I}}{\partial \mathrm{y}}\right) & \left(\frac{\partial \mathrm{I}}{\partial \mathrm{y}}\right)^{\mathbf{2}} \end{bmatrix} \tag{41}$$

If α and β are the two eigenvalues of **M**, then a corner will be found only when both α and β have large positive values. To reduce the calculation cost, the following equation will be used instead of finding the eigenvalue value:

$$\mathrm{R} = \mathrm{Det}(\mathbf{M}) - \mathrm{kTr}(\mathbf{M})^2 \tag{42}$$

Once the feature points are selected for the initial frame, the KLT algorithm (Tomasi and Kanade 1991; Lucas and Kanade 1981; Shi and Tomasi 1994) was adopted to track the features for the entire video. The intensity of the current frame, $J(\mathbf{x})$, can be expressed by using the intensity of the previous frame, $I(\mathbf{x})$, as below, by assuming a small amount motion by the feature.

$$J(\mathbf{x}) = I(\mathbf{x} - \mathbf{d}) = I(\mathbf{x}) - \mathbf{g} \cdot \mathbf{d} \tag{43}$$

, where **d** is the displacement vector between the two frames, and the gradient vector $\mathbf{g} = (\frac{\partial \mathrm{I}}{\partial x}, \frac{\partial \mathrm{I}}{\partial y})$.

The residue ϵ for the given window, including the feature point and the neighborhood surrounding it, can be defined by following equation.

$$\begin{aligned} \epsilon &= \int [I(\mathbf{x} - \mathbf{d}) - J(\mathbf{x})]^2 w d\mathrm{A} = \int [I(\mathbf{x}) - \mathbf{g} \cdot \mathbf{d} - J(\mathbf{x})]^2 w d\mathrm{A} \\ &= \int (h - \mathbf{g} \cdot \mathbf{d})^{\mathbf{2}} w d\mathrm{A} \end{aligned} \tag{44}$$

, where w is a weighting function, and $h = I(\mathbf{x}) - J(\mathbf{x})$.

To minimize the residue, the equation above can be differentiated with respect to $\mathbf{d}$ and set the result equal to zero as following.

$$\frac{d\epsilon}{d\mathbf{d}} = \int (h - \mathbf{g} \cdot \mathbf{d})\mathbf{g} w dA = 0 \tag{45}$$

Finally, the displacement vector $\mathbf{d}$ can be obtained by following equation.

$$\mathbf{d} = \mathrm{G}^{-1}\boldsymbol{e} \tag{46}$$

, where $\mathrm{G} = \int \boldsymbol{g}\boldsymbol{g}^T w dA$, and $\mathbf{e} = \int (I - J)\boldsymbol{g} w dA$

For each feature point $\mathbf{x}_i$, the displacement vector $\mathbf{d}_i$ was calculated by using the KLT algorithm that was discussed in the previous section. If the object is rigid and the displacement of the object is parallel to the camera, the displacements from the same object should be all equal. However, the feature points in the object will not have same displacements in reality, for several reasons. First, the noise in the image will cause some error toward the displacement. Second, if the structure exhibits out-of-plan behavior, the displacements vector will result in different values due to the projection. Last, and most important, some of the feature points might not be visible in the object. The selected region of interest might include feature points from other objects, and might be tracking the wrong points. This error, in both cases, is negligible and inevitable, but the latter issue should be removed in order to achieve accurate displacement measurement.

In this framework, the MLESAC estimator (Torr and Zisserman 2000) was used to remove the outlier displacement that had not been measured for the object, and to obtain the geometric transformation matrix to minimize the noise and out-of-plan error. The motion of the ideal feature points $\underline{\mathbf{x}}$ (without any noise) between two consecutive frames can be expressed as in the equation below:

$$\underline{\mathbf{x}}_i' = \mathbf{H}\underline{\mathbf{x}}_i \quad (47)$$

, where $\underline{\mathbf{x}}_i$ is the set of the homogenous images points of the features in the first frame, $\underline{\mathbf{x}}_i'$ in the next frame, and **H** is the transformation matrix considering affine transformation. However, because of these noises, the result of the above equation will not be exactly equal; instead, it will display some residuals. Here, the minimum number of random sample of correspondences is selected, and is estimated using the minimal set. By defining the residual error as the sum of the distance between the original and transformed feature points, and using the Expectation Maximization (EM) algorithm, the final transformation matrix and the inlier can be obtained. The inliers will be tracked for the next frame by using the KLT algorithm, and will be repeated through the last frame. The displacement of the object can be measured in pixels by tracking the center of the ROI.

System Identification

Once the dynamic displacement of the structure is obtained, the next step is to analyze the dynamic characteristics of the structure by using system identification. System identification is a technique for estimating a mathematical model that represents the physical structure. In this task, Eigen system Realization Algorithm (ERA), as proposed by Juang and Pappa (1985), is used to estimate the system with known input.

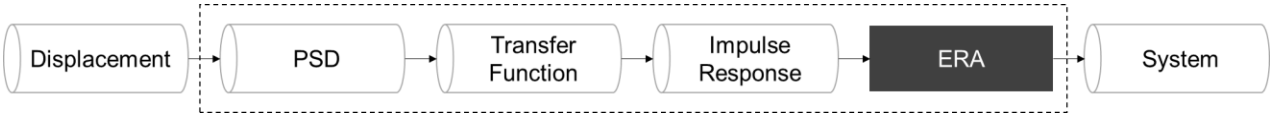


Figure 42: Flowchart for system identification.

Four steps must be done to perform system identification using the ERA method (Figure 42). First, based on the displacements determined from the tracking algorithm, the Power Spectral Density (PSD) function can be obtained. Next, the transfer function is obtained by dividing the output PSD function with the input PSD function. Finally, the Impulse Response Function (IRF), which will be used as the input of the ERA, can be obtained by applying the Inverse Fourier Transform.

ERA uses the IRF to construct the Hankel Matrix, which represents the data structure for the Ho-Kalman algorithm (HO and Kálmán 1966). The Hankel matrix is decomposed by singular-value decomposition to determine the order of the system. Finally the system can be obtained by finding the Eigen solution of the realized state matrix.

However, in the site experiment, measuring the input of the system is not trivial. Therefore for output-only identification, the Natural Excitation Technique (NExT) (James III, Carne, and Lauffer 1993), is used together with ERA. Cross-Power Spectral Density (CPSD) was calculated instead of PSD, and the cross-correlation matrix, which will be used as the input for ERA, can be obtained by applying the Inverse Fourier Transform.

5.2 Case Study: System Identification for the 6DOF Building Model

Test Setup

An experiment was designed to verify the proposed system identification framework. Figure 43 shows the different components of the experiments, in which the analyzed structure is a six-story model with equally distributed masses that are connected through elastic springs. The model is fixed on a uni-directional shaking table with a maximum displacement stroke of five inches. The model response is recorded using two different instruments, which are described below.

Cameras: two commercial cameras were used to record the displacement output of the model and the shaking table. These cameras were oriented perpendicular to the motion axis, as shown in Figure 43. The first camera is a GoPro Hero3 Black edition, which can record with a frame rate of 120 fps and 720p resolution. The other one is the LG G3 smartphone camera, with recording rate of 30 fps and a resolution of 1080p. In order to improve the accuracy of the displacement detection process, a white background was fixed behind the tested model. Moreover, camera calibration was performed to the two cameras prior to conducting the experiments.

Reference sensors: six accelerometers were attached to the different stories of the structure in addition to one accelerometer attached to the shaking table that measured the input accelerations. The accelerometers were connected with wires to a data acquisition system (DAQ) called VibPilot, which in turn was connected to a computer to record the response. For each sensor, the sensitivity was 100 mV/g, with a weight of 0.95oz, while the sampling frequency was adjusted to 1024 Hz, with the anti-aliasing filter turned on.

The purpose of this experiment was to verify the proposed framework using typical commercial cameras to capture the structural displacements at wide frequency ranges, which would be used for the system identification procedure. The identified system would then be compared to the accelerometers results as a reference solution. In order to excite the different structure modes of vibration, a band-limited white noise (BLWN) was adopted as an input motion in this experiment, which can be designed to cover a wide range of frequencies.

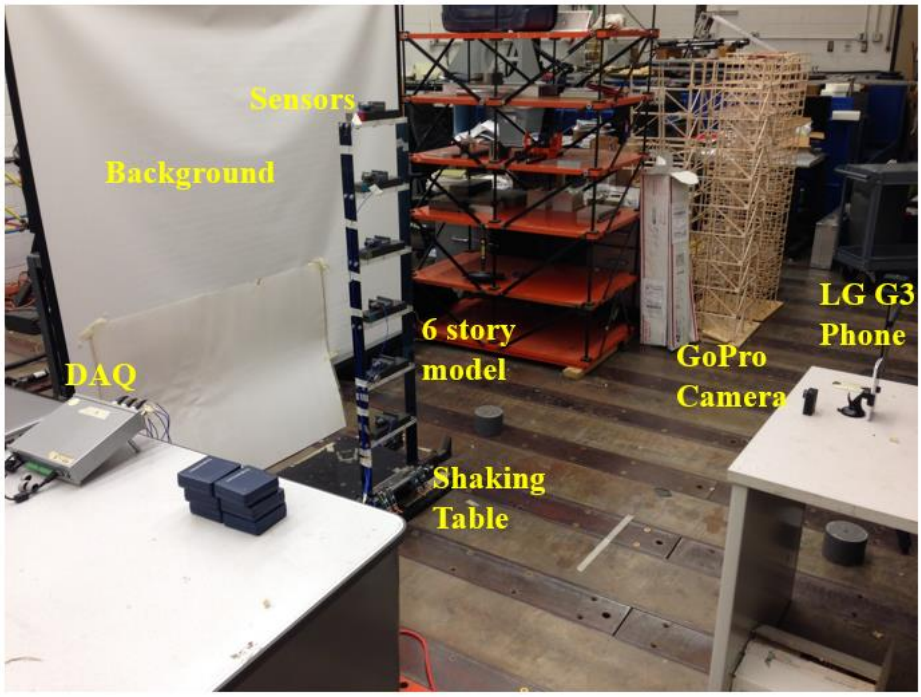


Figure 43: Different components for the validation experiment.

Displacement Results

Based on the described test setup, the experiment was executed and the recorded videos were analyzed. The KLT tracker was used to detect and track the features from the model during the analysis. The procedure was adopted to determine the displacement from the recorded video, and can be summarized as follows:

- The video was calibrated to remove the optical distortions for accurate displacement measurement.
- The user specified the region of interest for each degree of freedom (DOF) in the model, as shown in Figure 44a. Afterward, Harris corner detection was applied to detect the features within each of the specified regions, as detailed in Figure 44b.
- The KLT algorithm tracked the detected features and determined their pixel coordinates throughout the subsequent frames in the recorded video, as in Figure 44c.

- The MLESEC algorithm was used to remove the outliers and calculate the rigid motion of the object. The determined coordinates were subtracted from the initial values to calculate the relative motion of the features.
- A scaling factor was calculated to transform the units of the displacements from pixels to a physical unit, which can be obtained by measuring a known length in the model. In this experiment, the scaling factor was 22 pixel/inch.
- A band-pass filter was applied to eliminate any unwanted noise that might have been induced during the testing process. In this experiment, the filter was designed as the Elliptic Filter, with a band-pass width of 1 to 120 Hz.

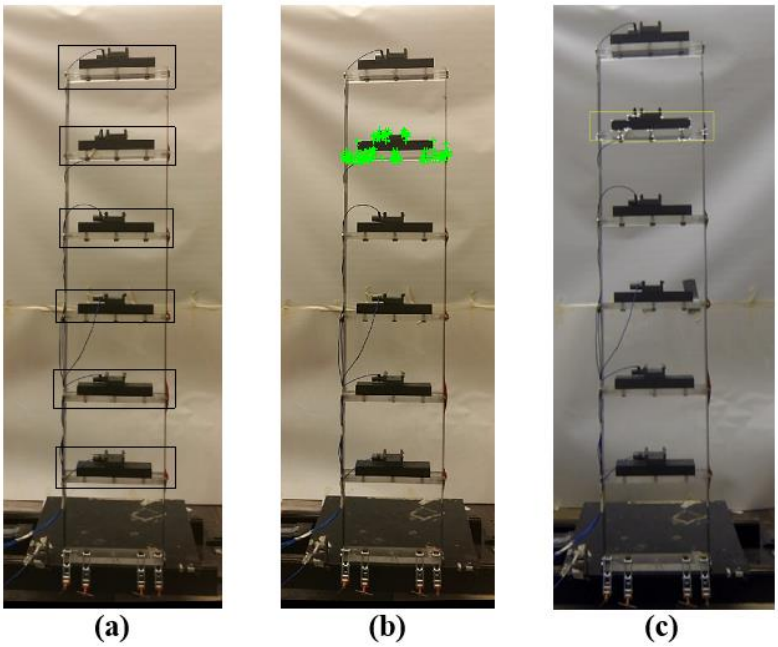


Figure 44: Feature detection procedure; (a) ROI selection (b) feature detection (c) feature tracking.

Using the proposed procedure, the filtered displacement response was plotted against the time series for the six stories in the analyzed model, as shown in Figure 45. In the context of these results, the model was subjected to the BLWN input, and a GoPro camera was used to record the video.

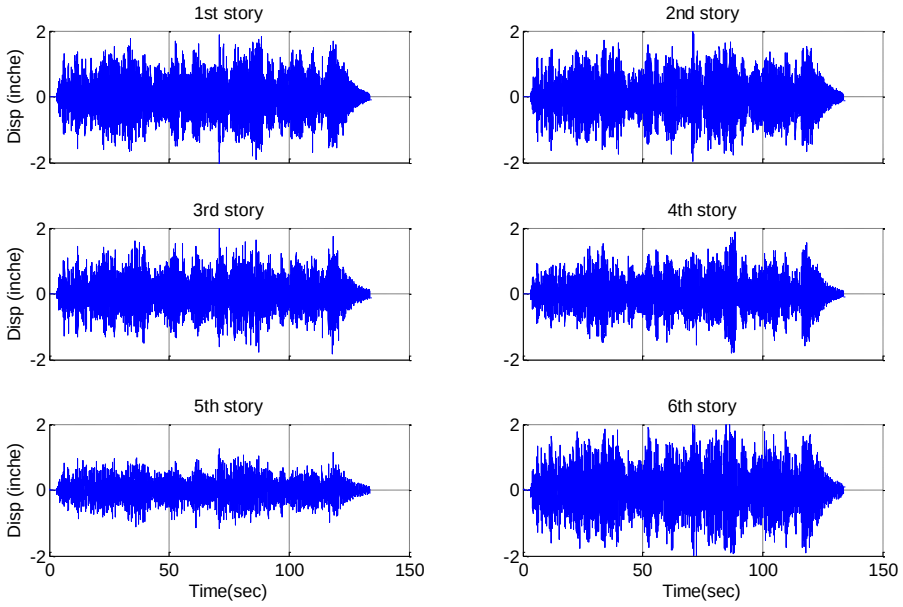


Figure 45: The BLWN displacement response for the six DOFs of the tested structure.

System Identification results

In this section the system identification technique was applied to the six-story model when subjected to the BLWN input motion. Two of the commercial cameras were used for this purpose, which were the GoPro Hero3 camera (120 fps) and the LG G3 smartphone camera (30 fps). The results were then compared with the reference accelerometers attached at each DOF of the model, as described in the test setup section. The accelerometers recorded the data at a rate of 1024 Hz to achieve a reliable system result relative to the both cameras. In addition, an anti-aliasing filter was applied through the data acquisition system to eliminate the effect of frequencies above the Nyquist

value. The state space matrix and the modal properties of the system were evaluated using ERA method, whereby the displacements determined from the two cameras and the accelerations measured using the accelerometers were utilized for that purpose. Figure 46 shows the identified transfer functions using the ERA method against the actual experimental records for the accelerometers, and the two cameras. The figure shows that the transfer function estimated using the ERA method follows a similar pattern to the experimentally evaluated records. The values identified using the accelerometers were more consistent than for the GoPro camera ones, which in turn were better than the LG G3 camera results. This behavior was expected, due to the difference in sampling frequency for the different measurement instruments.

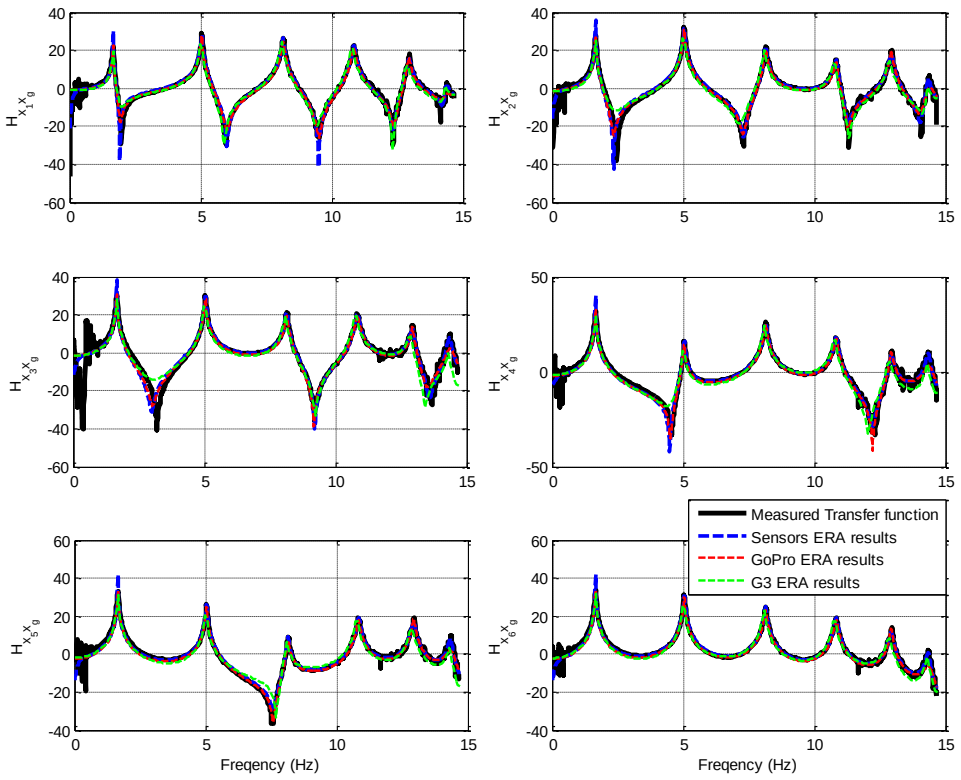


Figure 46: Comparison of ERA-estimated transfer functions using different measuring instruments.

According to the state space matrix identified for the tested model, the natural frequencies and the mode shapes were able to be determined. Table 7 shows the natural frequencies using the different instruments. For these results, the accelerometers were assumed to provide the most reliable results. The cameras were able to capture the natural frequencies within errors of approximately 0.2% and 0.7% for the GoPro and LG G3 cameras, respectively. It can be observed that the GoPro camera achieved more accurate results compared to the LG G3 camera for the six natural frequencies. These results are due to the higher recording frame rate of the GoPro camera, which provides more reliability for the estimated system. On the other hand, Figure 47 compares the mode shapes of the structure, estimated using the ERA method. As can be observed from the figure, the mode shapes followed the same pattern and relatively close magnitudes for the six DOFs—except for the sixth mode, which could not be captured accurately by LG G3, due to its low frame rate. Finally, the first five mode shapes using LG G3 (1080p) were better than for the GoPro (720p) ones. This result can be justified because higher camera resolution allowed the tracking algorithm to more precisely detect the vibration amplitude. Therefore, we can conclude that the higher frame rate resulted in better accuracy for the temporal domain results (e.g., natural frequency), and the resolution of the pixel provided better results in the special domain (e.g., displacement amplitude, mode shape).

Table 7: The natural frequencies for the different measurement cases.

	Natural Frequencies (Hz)			Error (%)	
Mode	Sensors	GoPro	LG G3	GoPro	LG G3
1	1.657	1.660	1.652	0.164	-0.323
2	5.038	5.038	5.004	-0.001	-0.684
3	8.138	8.143	8.086	0.065	-0.639
4	10.833	10.834	10.759	0.004	-0.689
5	12.930	12.931	12.850	0.008	-0.623
6	14.339	14.368	14.303	0.205	-0.252

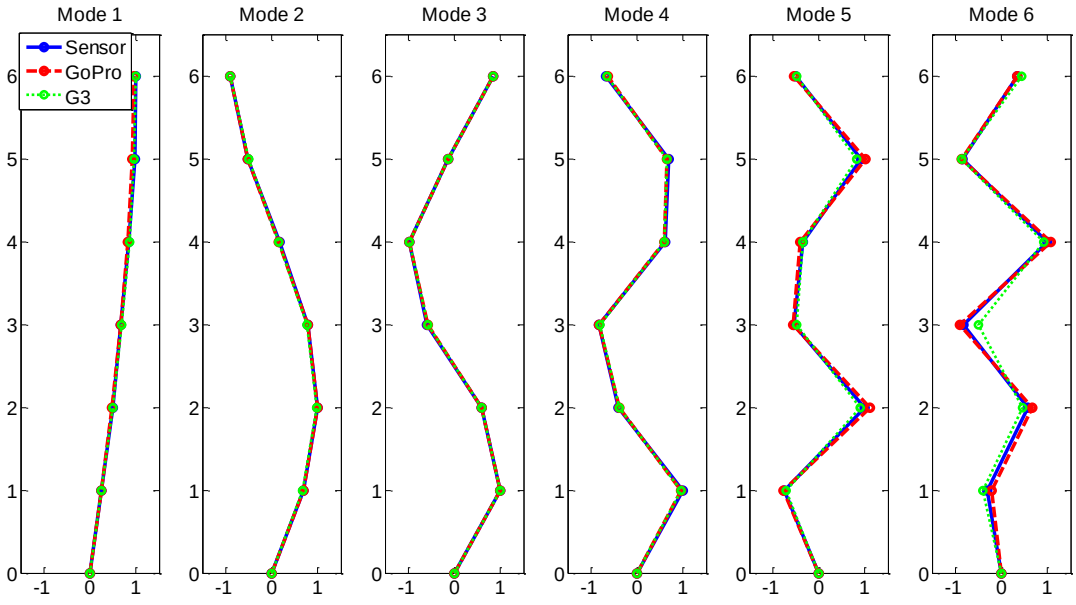


Figure 47: Comparison of the mode shapes estimated using the ERA method.

5.3 Discussion

This task presented a new approach toward identifying the dynamic characteristic of structures by using commercial cameras without employing any target marker. Despite promising results, there are several research challenges which remain open for future investigation. These challenges include:

(1) Selecting the Region of the Interest Box: Selecting the region of the interest is the only procedure in which human input is required. However, this procedure can significantly affect the quality of the result. First, if the ROI is too large (Figure 48.a) then numbers of the feature points might exist outside of the object of interest. If one of the objects, other than the object of the interest, has more feature points to track compared to the number of features in the object of the interest, the feature points might be removed by being consider outliers. On the other hand, if the ROI is too small (Figure 48.b), then the number of the feature points detected inside the box will be too small and might be lost during the process. Finally, the features in the object will have the same motion if the object of the interest is not rigid (Figure 48.c). The transformation matrix discussed in Section 2 is considers affine transformation. If the object has multiple degrees of freedom, then only one mode—the one that has largest numbers of features—will be tracked, and other modes will be neglected.

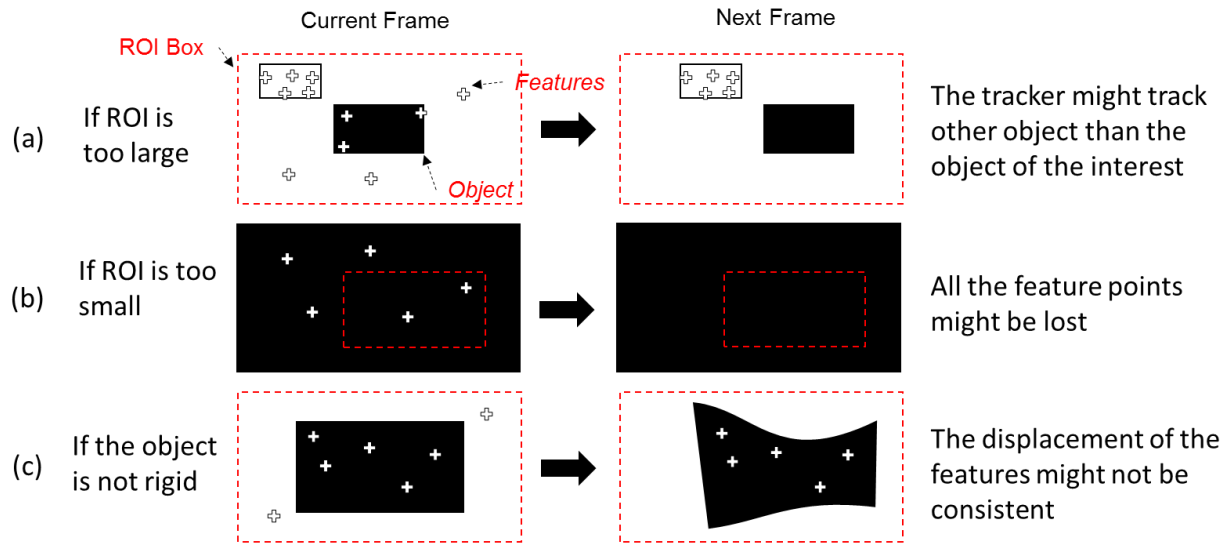


Figure 48: Issues with selecting the region of the interest box.

By considering the issues discussed above, the region of the interest box should be selected carefully by following directions. First, the size of the ROI should be selected so that the number of features in the object of interest should be greater than the number of features in any other object in the ROI. Second, the size of the ROI should be selected so that at least three feature points (i.e., six values) exist in the ROI. A minimum of three set of points are required to obtain an affine transformation matrix; however, considering noise and the imperfections inherent in these processes, more than three points is recommended. Finally, the object of interest should be firmly rigid.

(2) Temporal Aliasing: Professional high-speed cameras are widely being used for the dynamic displacement measurement for structural system identification. However, these cameras are not only expensive but also heavy. On the other hand, commercial cameras, such as smartphone cameras, or GoPro cameras, are inexpensive and convenient to use—hence, they are commonly carried by people most of the time in their daily activities. Even with these advantages, however,

there is the issue of utilizing commercially-provided cameras for the proposed system—namely, temporal aliasing. Compared to high-speed cameras, the commercial cameras usually have a low frame rate (ex. 30 fps or 60 fps). When structure response has a frequency of higher than half of the frame rate, the measured displacement will contain aliased information from higher frequencies. The aliasing effect can be usually removed in a wired/wireless sensor system by having an Anti-Aliasing (AA) filter. However, temporal aliasing cannot be removed in vision-based systems, because the images are already aliased when taking digital images. Therefore, the maximum frequency of the structure response that a camera can measure will be half of the frame rate (fps or Hz). If a high frequency response that exceeds half of the framerate exists, a camera with higher frame rate will be required; otherwise, the measured displacement will be incorrect.

(3) Inaccurate Frame Rate: Another issue with the frame rate of commercial cameras is that they are not only low, but also inaccurate and not reliable. The frame rates in their specifications are not accurate enough to be used with the proposed system. For example, the LG G3 cameras display a frame rate of 30 fps in the specification. However, the actual frame rates in the meta-data were 29.45 fps, which was off by about 2%. The error in frame rate can cause significantly incorrect results in the natural frequency of the structure; therefore, the frame rate should be accurately obtained with the metadata before using the proposed system.

5.4 Summary

This chapter proposed a target-free approach for a vision-based system identification using commercial cameras. The method includes three procedures: (1) camera calibration; (2) a vision-based, dynamic displacement measurement, which extracts the features with Harris comer detection, tracks the features with the KLT tracker, and removes the outliers using the MLESAC

algorithm; and (3) system identification using ERA. Unlike the vision-based system identification methods that require target markers and professional cameras, the proposed method enables one to obtain the dynamic characteristics of structures using commercial cameras without attaching any artificial targets to the structures. To validate the proposed method, the response of a six-story model situated on a shaking table was measured by two commercial cameras together with a reference accelerometer. The experimental results showed that in employing the proposed method it is possible to identify the natural frequency and the mode shape with a reasonable level of accuracy, Furthermore, the author discussed major issues with regard to the proposed system, including ROI selection, aliasing, and inaccurate frame rates.

CHAPTER 6 STRUCTURAL MONITORING USING UAVS

As discussed in Section 2, the infrastructures that already have wired or wireless SHM systems installed could easily be monitored; however, most infrastructures, especially in developing countries, do not have such SHM systems installed. Once a disaster happens, installation of the sensors is not trivial; thus, the conventional wired and wireless SHM systems cannot be easily deployed. Therefore, a new technique for SHM system using cameras and UAVs is proposed.

A few researchers have conducted studies on vision-based structural health monitoring; however, some issues still remain to be resolved before these techniques can be used together with UAVs. First, many previous vision-based SHM methods require a target marker. Installing target markers on infrastructure can create significant risks, especially when the structure has not yet been inspected. Second, professional cameras are heavy and expensive. To install professional cameras, the UAV should be large enough to accommodate the weight of the camera and would be inefficient from an energy standpoint. Third, the post-processing procedure requires experts with structural engineering or professional knowledge. To be used in a post-disaster situation, a condition assessment should be done automatically. Finally, and most importantly, the motion of the camera had not been considered. In order to utilize aerial images, the motion of the camera and the UAV should be calculated and removed from the video.

Therefore, the following steps are proposed for overcoming the limitations discussed above. First, the non-target-based SHM system will be put forward. Next, a deeper study using a commercial UAV for SHM will be discussed. A frame-rate issue and the relationship of the resolution of the camera to the accuracy of system identification will be considered. Third, an

automatic procedure including the region of interest selection will be proposed. Finally, compensation for the motion of the camera will be discussed.

6.1 Overview

Figure 49 shows the overview of the proposed method. The underlying pipeline is composed of two main components. The first is a target-free measurement of the displacement of the bridge relative to the UAV; this approach does not require a target to be placed on the structure. The second is removing the UAV motion to yield the absolute motion of the bridge. Each main component has sub-components.

For a target-free displacement measurement, the first step is to conduct camera calibration to remove radial distortion and obtain the intrinsic camera matrix. Next, select the natural features from the bridge. Finally, track the features using Optical Flow to obtain the relative displacement of the bridge with respect to the UAV motion (Yoon et. al, 2016).

While the relative motion of the bridge is obtained from the features of the bridge, the motion of the camera induced by UAV motion is estimated by using background features. Once the camera motion is estimated, the absolute displacement of the bridge can be obtained by re-projecting the foreground features from the corresponding point of view. This method is described in more detail in subsequent sections.

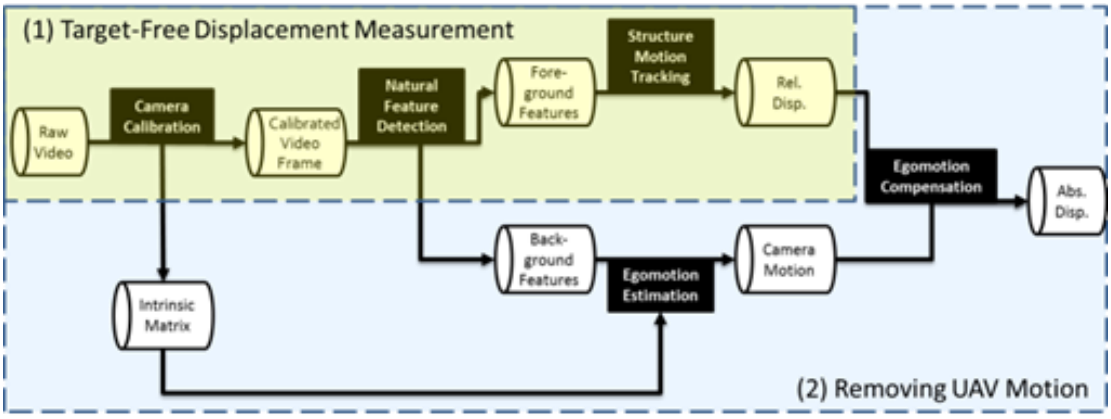


Figure 49: Overview for Displacement Measurement using UAV.

6.2 The Target-Free Displacement Measurement

The first step in the proposed approach is camera calibration. Modern consumer-grade camera lenses have improved dramatically in recent years. However, the recent trend toward small, lightweight, low-cost, and high-definition action cameras, such as GoPro or those used in commodity smartphones, often employ wide-angle lens. These lenses increase the field-of-view by intentionally introducing significant radial distortion. To remove this distortion and obtain accurate displacement measurements using consumer-grade cameras, performing camera calibration is necessary. Another reason for calibrating camera is to obtain the Intrinsic Matrix, which includes unique parameters of the camera lens independent of the location of a particular scene. These parameters are essential for the latter part of the proposed method when removing the UAV motion. The camera calibration process can be conducted by taking pictures of known geometry points (e.g., girders or joints) from different point of view (Figure 50).

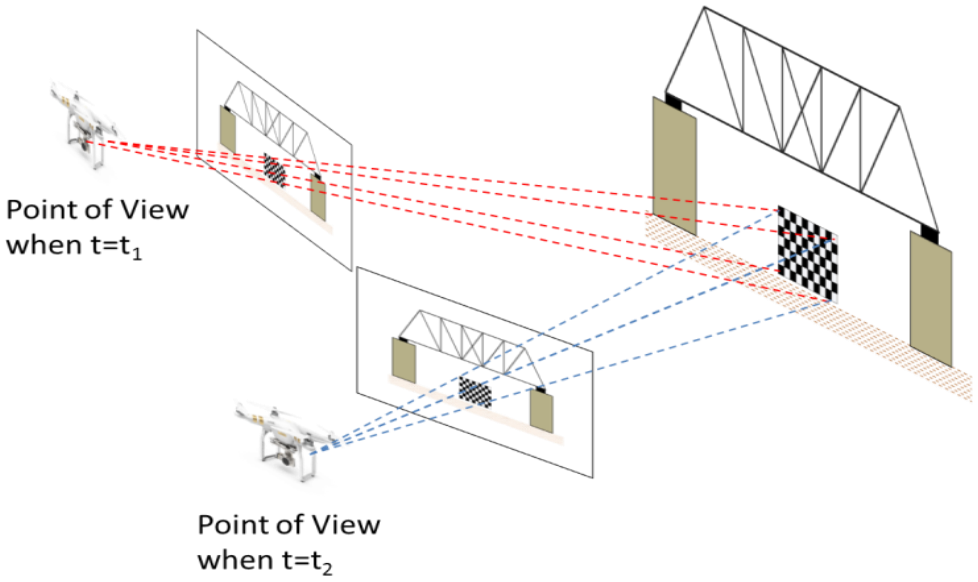


Figure 50: Camera Calibration and Ego-motion Estimation.

Once the camera is calibrated and image distortion is removed, the dynamic response of the structure relative to the UAV is determined by analyzing the video, frame-by-frame. To achieve reliable tracking, distinct (i.e., highly discriminative and salient) features are selected from the objects of interest. These features should be invariant to changes in illumination, scale, and pose (rotation and affine), and they should characterize the local proximity of the points of interest. Several feature detection methods can be used for this purpose. In this work, the corner detection method suggested by Harris and Stephens (Harris and Stephens, 1988) is used to extract features within the region on interest (ROI) in the initial video frame. The Harris corner detection method detects points with a large value of the weighted sum of the intensity differences squared. Points with this characteristic are called corner points. These Harris corner points are effective feature points used to track in methods such as the KLT algorithm.

After the features are selected for the initial frame, the KLT algorithm is adopted to track the point features for the entire duration of a video. Assuming a small amount of motion between consecutive frames, the optical flow vector can be obtained by minimizing the residual of the difference between the intensity of the previous and the following frame. The accuracy of the displacement result can be enhanced by applying Image Pyramids and outlier removal methods such as RANSAC. Finally, the displacement vector with relative to the UAV will be obtained.

6.3 Removing UAV Motion

To obtain the degree of absolute bridge displacement, the motion of the UAV should be estimated and removed. A simple method for determining UAV motion is to track the stationary objects in the background. This approach will not guarantee accurate results due to two primary issues. First, the motion of the camera in the UAV is not pure translation, but does include rotation. Also, even if the UAV moves in pure translation, the background features being tracked are not always in the same plane as the bridge. Therefore, this paper proposes a new method for estimating UAV motion.

The 3D motion of a camera within an environment is often called ego-motion, which is the location and the orientation of the camera in the UAV. The ego-motion can be estimated by tracking the background points together with the intrinsic matrix obtained in the camera calibration step. The translation vector $\boldsymbol{t}$ and the rotation matrix $\boldsymbol{R}$ of the camera can be calculated by using the Eight-Point Algorithm (Hartley, 1997) as shown in Figure 50.

The estimated parameters (relative rotations and translations) inherently involve the issue of noise; therefore, jointly optimal parameters estimates are needed, which is referred to as a bundle adjustment (Triggs et. al, 1999). In typical Structure from Motion (SfM) bundle adjustment

formulations, the parameters are optimized by adjusting intrinsic parameters, extrinsic parameters, and triangulated points, as in Equation 48 below,

$$min \sum_{i=1}^{n} \sum_{j=1}^{m} \left\| f(P_{ij}) - X_{ij} \right\|^2 \tag{48}$$

where, X_{ij} represents the observed image points, $f(P_{ij})$ is the projected point using the estimated Rotation and Translation in the ith feature point of the image j.

This bundle adjustment can increase the accuracy of the camera pose parameters by use of the optimization process. However, the bundle adjustment for UAV motion estimates has two major issues: (1) parameters (rotations and translations) are similar to each other, which results in significant error, (2) it takes long time to optimize the parameters. Furthermore, when the motion of the UAV is small compared to the distance between the target and the UAV, then the result of the estimated poses is also inaccurate. To address these two issues, constrained bundle adjustment (Karsch, Golparvar-Fard, and Forsyth 2014) was implemented for each single frame referenced, with additional frames taken from the large motion of UAV (Figure 51). This approach optimizes a single set of rotations and translations parameters referenced to large UAV motions, which result in increased accuracy within a shorter calculation time:

$$min \sum_{i=1}^{n} \| g(P_i) - X_i \|^2 \tag{49}$$

where, X_i is the observed image points, $g(P_i)$ is the projected point using the estimated Rotation and Translation calculated with reference images in the *i*th feature point.

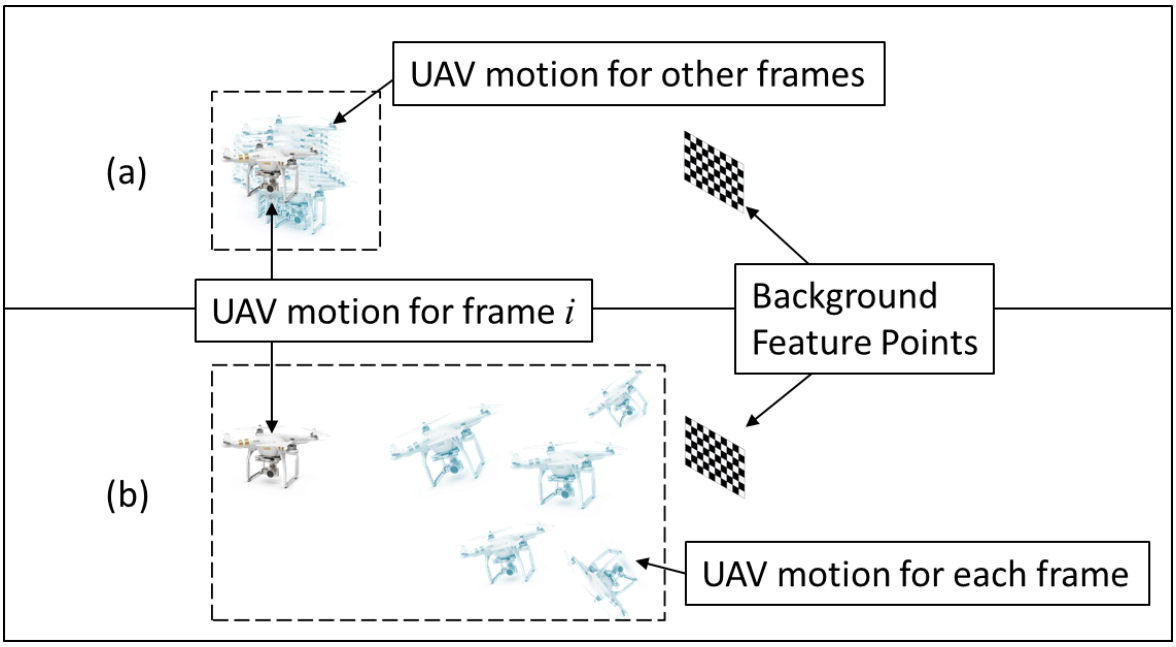


Figure 51: Bundle Adjustment for (a) frames between UAV motion, (b) *i*th frame with additional calibrated frames.

Once the location and the orientation of the camera in the UAV is determined, the camera projection matrix ***C***, which maps 3D points among world coordinates into an image coordinate, can be calculated (Equation 50). Using the pinhole camera model, the equation between the 2D projected points (x,y) and the world coordinate points (X,Y,Z) can be written as Equation 51. By neglecting the out-of-plane (transverse to the longitudinal axis of the bridge) displacement of the bridge, the world coordinate of the feature points in the bridge can be estimated by using Equation 52. The displacement vector obtained from this process will be the absolute displacement of the UAV.

$$\boldsymbol{C_{4\times3}} = \boldsymbol{K_{3\times3}}[\boldsymbol{R_{3\times3}} \quad \boldsymbol{t_{1\times3}}] \tag{50}$$

$$s\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \boldsymbol{K_{3\times3}}[\boldsymbol{R_{3\times3}} \quad \boldsymbol{t_{1\times3}}]\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \boldsymbol{C_{4\times3}}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{51}$$

$$\begin{bmatrix} X \\ Y \\ s \end{bmatrix} = \begin{bmatrix} C_{11} & C_{21} & -x \\ C_{12} & C_{22} & -y \\ C_{13} & C_{23} & -1 \end{bmatrix}^{-1} \left\{ -\begin{bmatrix} C_{31} \\ C_{32} \\ C_{32} \end{bmatrix} Z - \begin{bmatrix} C_{41} \\ C_{42} \\ C_{43} \end{bmatrix} \right\} \tag{52}$$

where, s is the scaling factor, $\boldsymbol{C}$ is the camera projection matrix, $\boldsymbol{R}$ is the rotation matrix, $\boldsymbol{t}$ is the translation vector, and $\boldsymbol{K}$ is the intrinsic matrix at each frame.

6.4 Case Study: Railroad Bridge Displacement Measurement using UAV under Train Load

Experiment Setup

The Federal Aviation Administration has placed considerable restrictions on flying a UAV in the field. Therefore, preliminary experiments were conducted in the Newmark Structural Engineering Laboratory at the University of Illinois at Urbana-Champaign to validate the proposed approach. The vertical motion of a pin-connected truss bridge owned by the CN Railroad near Rockford, Illinois, which is subjected to revenue-service traffic, was measured and reproduced on a servo-hydraulic motion simulator (Figure 52). The DJI Phantom 3 Professional (Figure 53) mounted with a 4K resolution (4096x2160) camera operating at 24 fps was selected for this experiment. The UAV recorded the video at a distance of 15 feet from the motion simulator, which

corresponded to the distance so as not to foul the track (Figure 54). To have a reference by which to assess the accuracy of the proposed method, a Krypton 3D measurement system (K600) with an accuracy of 0.02mm was installed.

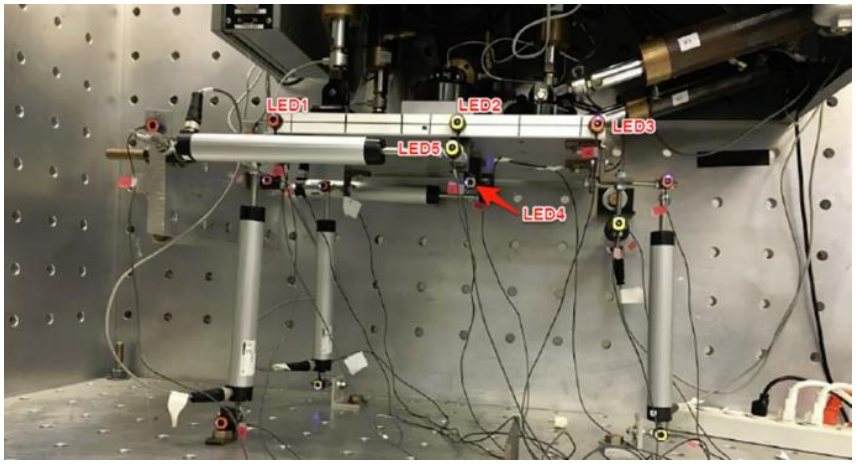


Figure 52: Motion Simulator with Krypton LEDs.

Figure 53: DJI Phantom 3 Professional.

Experimental Result

Figure 55 shows the relative displacement of the UAV with respect to the bridge that was estimated from the video taken with the UAV. When compared with the reference measurement of the

simulated motion of the bridge, very little information about the motion of the bridge appears to be contained in this estimate.

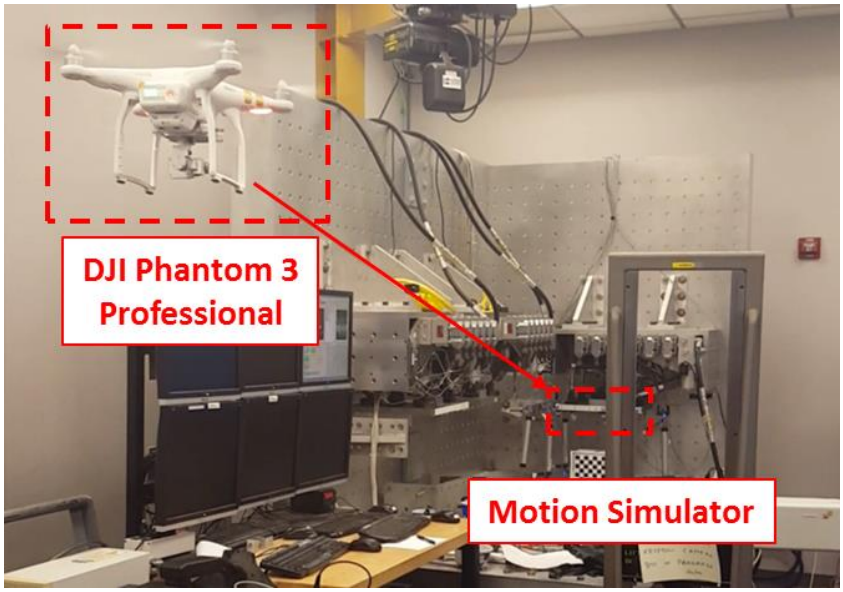


Figure 54: Experiment Setup using LBCBs.

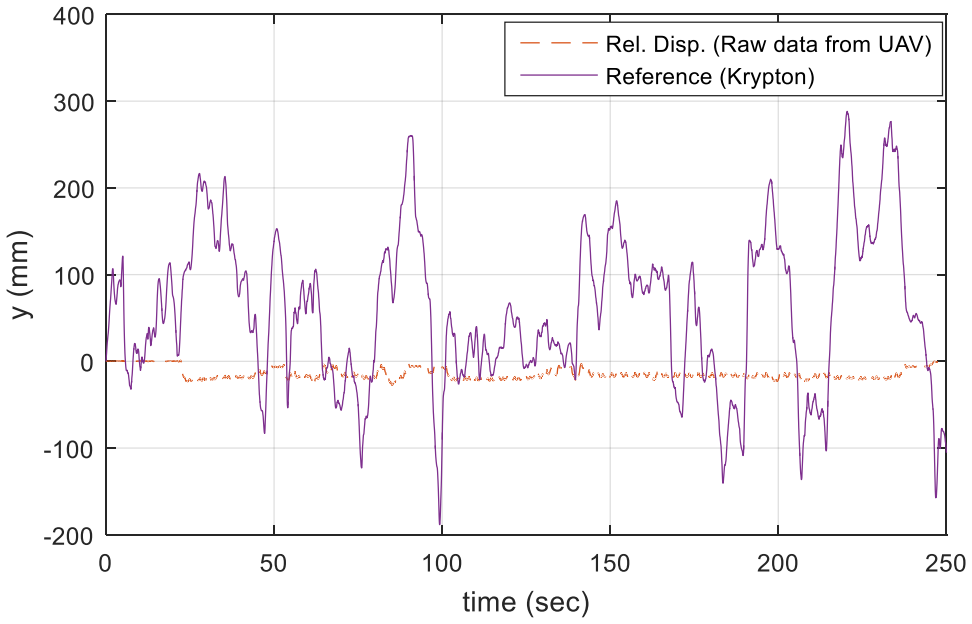


Figure 55: Relative Displacement Respected to UAV.

Using the proposed ego-motion estimation method, the 6 degree-of-freedom motion of the UAV was estimated as shown in Figure 56. As is demonstrated in this figure, the motion of the UAV is complex, containing significant translations and rotations.

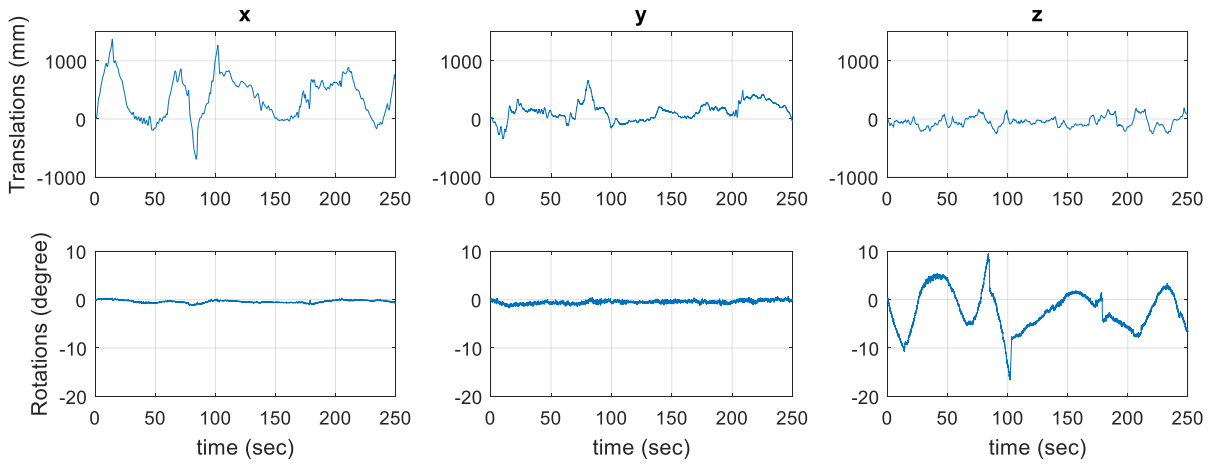


Figure 56: 6DOF Motion of the UAV.

The camera projection matrix, which includes information regarding both translation and rotation, was calculated. Figure 57 shows the absolute displacement of the bridge that was determined by using this camera projection matrix. The displacement determined using the proposed method matched well with the simulated vertical motion of the railroad bridge. The RMS error was 2.14 mm, corresponding to 1.2 pixel of resolution. These laboratory results demonstrate the potential of the proposed method.

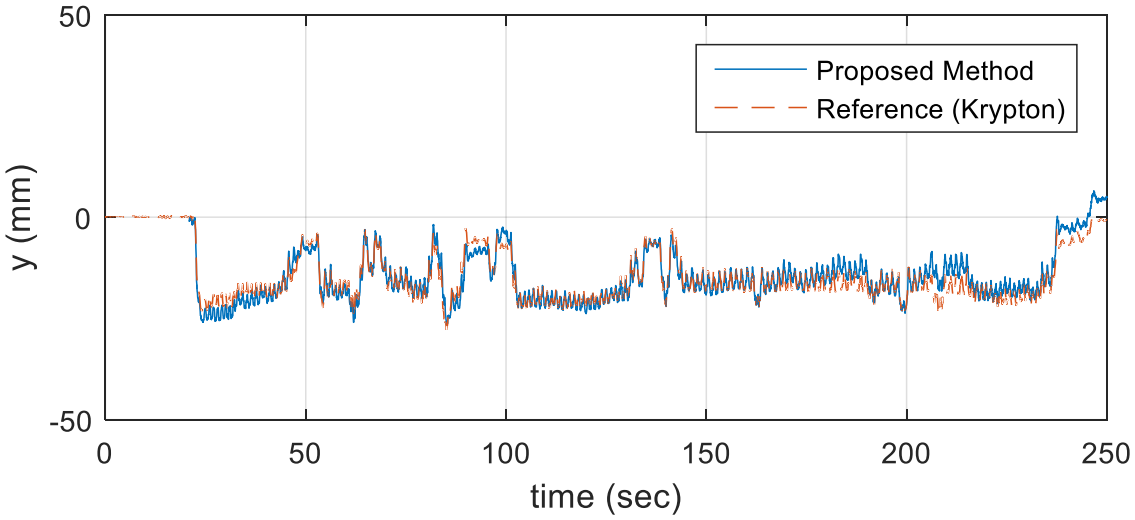


Figure 57: Absolute Displacement using Proposed Method.

While the proposed method showed promising results, there are some issues that can be addressed to enhance the current approach. The error of the proposed method was mostly due to the large motion of the UAV. It can be shown that the error (difference between the red and blue lines in Figure 57) of the proposed method is related to the magnitude of the motion in Figure 56. Indeed, the correlation coefficient for the time window when there is a sharp peak in rotation (e.g. time 83 ~ 85 sec) was 0.88 while the correlation coefficient for the entire data was only 0.168. The proposed method can be enhanced by minimizing the motion of the UAV with more precise control, or even by analyzing only the data with low level motion.

6.5 Summary

While the laboratory test was conducted successfully, many uncertainties still exist in the field such as wind and light conditions. To validate the proposed method to be used in practice, on-site field tests are being planned at TTCI. Also, for better accuracy, a zoom lens and the use of inertial measurement sensors are being considered.

CHAPTER 7 CONCLUSIONS AND FUTURE STUDIES

7.1 Conclusion

The research outlined in this dissertation has provided methodologies for post-disaster response and structural health monitoring using sensing technologies for commercial off-the-shelf devices. Smartphones were used to localize trapped victims and estimate their statuses; dead-reckoning and computer vision methods were used to track first responders and localize important events; and commercial cameras and UAVs were used to measure the displacement of civil infrastructures for the purpose of health monitoring.

Smartphones were also used to estimate the location and status of the trapped victims. To localize victims with room-level accuracy, a WLAN-based indoor localization technique was adopted, together with the use of a Naïve Bayes Classifier. A WiFi signal captured by smartphones was used to classify the location of the user by considering the directionality obtained from the magnetic field sensor as embedded in a smartphone, and the Naïve Bayes classifier was adjusted so that the method could be used robustly even in a disaster situation in which some of the wireless access points might have been damaged. To estimate the status of the victim, multi-metric sensors embedded in the smartphone (e.g., an accelerometer, a gyroscope, a magnetic field sensor, and a pressor sensor) were used to classify activities taking place during the disaster scenarios.

Dead-reckoning methods were utilized to track first responders and localize important events (e.g., fires, structural hazards, etc.). Because the WLAN-based indoor localization can provide solely room-level accuracy, a dead-reckoning system using the multi-metric sensors contained in smartphones was proposed. To determine the initial location, a landmark localization module was developed using a machine learning technique. To reduce the drift error of the

localized event, a probabilistic model for a smartphone-based dead-reckoning system was proposed. The error of the event location was reduced by applying the Forward-Backward approach. Furthermore, the proposed method enabled the researcher to remotely localize the hazardous events, simply by taking two or more picture while being tracked by the dead-reckoning system.

The target-free, vision-based method for structural health monitoring was also presented for system identification purposes, using inexpensive consumer-grade cameras. This method has three procedures: (1) camera calibration; (2) a vision-based dynamic displacement measurement, which extracts features using Harris comer detection, and tracks the features employing a KLT tracker, then removes the outliers with the use of the MLESAC algorithm; and (3) system identification using the ERA. The proposed method enables the dynamic characteristics of structures to be derived with consumer-grade cameras without attaching any artificial targets to the structures. To validate the proposed method, the response of a six-story model combined with a shaking table was measured by two consumer-grade cameras together with a reference accelerometer. The experimental results show that proposed method has potential for identifying the natural frequency and mode shape, with reasonable levels of accuracy. The practical considerations and limitations of the proposed system including ROI selection, aliasing, and inaccurate frame rate also were discussed.

Finally, utilizing unmanned aerial vehicles (UAVs) for measuring the displacement of structures was presented. The proposed method combined a target-free, vision-based method to measure the relative displacement to the camera, together with the 6 DOF camera pose estimation to accurately estimate the absolute displacement with a non-stationary camera installed in UAVs. To validate the proposed method, a simulated laboratory test was conducted by reproducing

railroad displacement under revenue train load. The result of this test showed an estimated RMS error of 0.08 inch at a standoff distance of 15 feet.

7.2 Future Studies

Adaptive Evacuation Route Optimization

While the location of the victims and their status, and the location of first responders and event locations can be determined with the methods proposed in this study, the evacuation route has not yet been optimized, taking into account all of this information. Therefore, providing an adaptive evacuation route will be a subject of future work. The evacuation route would be optimized based on the location of victims, their status, and other environment conditions. A preliminary study has been conducted utilizing Dijkstra's Algorithm and Ant Colony Optimization (Figure 58).

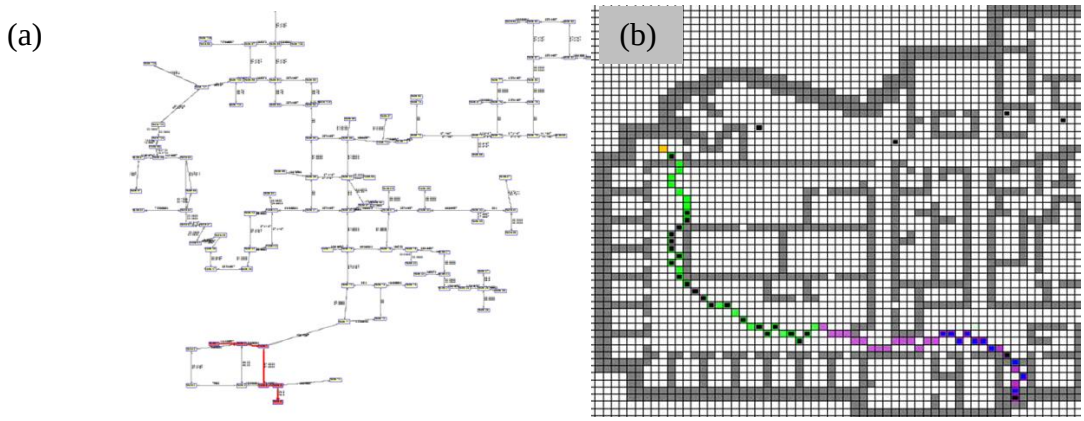


Figure 58: Evacuation Route Optimization using (a) Dijkstra's Algorithm, (b) Ant Colony Optimization.

Providing Solutions for Non-Smart Buildings

The proposed methods in this study rely on several assumptions that can be only applied to smart buildings. First, battery backed-up wireless access points were assumed to have been installed in the smart buildings. Second, fingerprinting data (e.g., a WLAN signal database) was assumed to have been stored in the access points prior to the disaster scenario occurring. Third, landmark location data and the digital floor map are also assumed to have been stored at the base station. All of this information was assumed to have been downloaded to victims' and first responders' devices upon arrival. However, there are many buildings that are not yet ready for this type of information, and therefore a new methodology will be needed for responding to victims in such buildings. For such non-smart buildings, research will be conducted in the future by using crowd sourcing techniques to collect these data.

Multi-metric Sensing using Additional Wearable Devices

The proposed methods for victim localization, assessment, and first responder tracking were all done using only smartphones. While at present smartphones are embedded with numerous sensors (e.g., an accelerometer, a gyroscope, a magnetic field sensor, a light sensor, a pressure sensor, and so on), additional wearable devices could provide an opportunity to produced more accurate and robust estimates. Therefore, research on utilizing commercial off-the-shelf wearable devices such as smart watches, smart glasses, smart bands, and an oculus for a post-disaster response system will be conducted in the future.

Post-Disaster Site Monitoring using Unmanned Ground Vehicles and Unmanned Aerial Vehicles

Finally, in the future I would like to bring the work that I have done with unmanned vehicles back full circle, again focusing on disaster response. Indeed, rescue robots (Figure 59) are starting to be used in the search and rescue process, and many researchers are working to leverage those robots for rescue operations (Casper and Murphy 2003). There is potential for applying the proposed dead-reckoning system to these rescue robots—tracking the robots and estimating a specific event location. In addition, a computerized vision-based approach will be used to automatically determine the condition of civil-infrastructures in post-disaster conditions by combining the proposed methods from structural health monitoring using UAVs.

Figure 59: Rescue Robot in Rescue Operation.

Feedback Control of UAV using Optical Flow

While the basic framework for measuring the displacement of acivil-infrastructure using UAVs has been presented in this work, there are still some issues remaining for practical application. One of the issues is controlling the UAV—that is, keeping the camera frame focused on the region of interest. When experiencing strong winds, especially without an expert UAV pilot hovering over the UAV so that the camera can focus on the region of interest, achieving such control is not a simple task. When the detected features in the reference frame depart from the frame, the method will not be able to track such features anymore. Therefore, I am currently working with a master's degree student, Jaeho Shin, to provide automatic feedback control of UAVs using Optical flow (Figure 60).

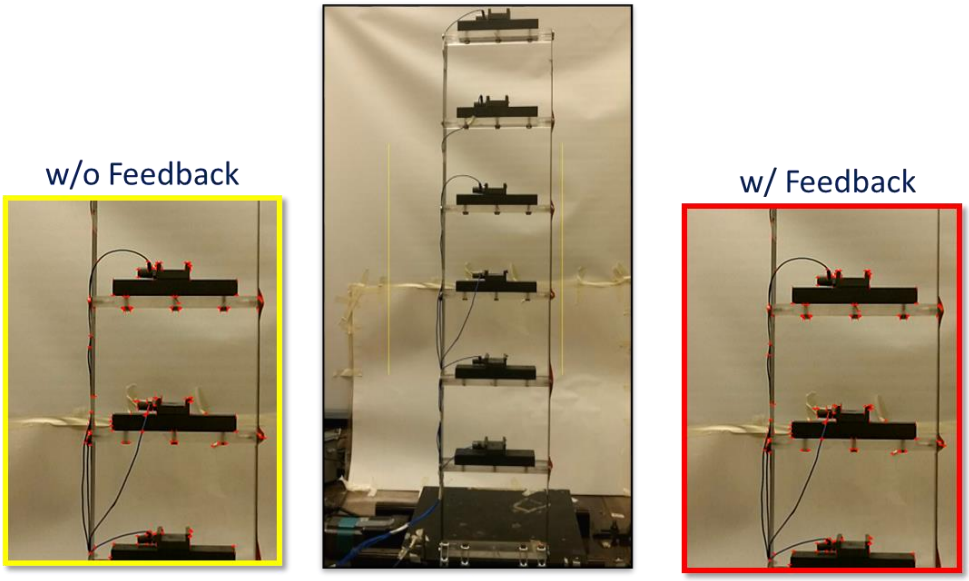


Figure 60: Feedback Control of UAVs using Optical Flow.

Other Applications of Structural Health Monitoring using Computer Vision

While Chapter 5 presented a System Identification method using stationary commercial cameras, and Chapter 6 presented structural monitoring applications using UAVs, the integration of these two works (system identification using UAVs) has not yet been conducted. The goal of the proposed application will be to automatically identify the system of the civil infrastructure by simply flying UAVs *around* the structures (Figure 61).

In addition to the System ID application, a structure point-of-view monitoring will be conducted (Figure 62). Instead of having cameras record the motion of the structure, cameras will be attached to the structure and record the view looking out from the structure. The 6 DOF motion of the camera pose will be estimated by tracking the background features, and structural health monitoring algorithms will be then applied.

Finally, the proposed method to estimate the 6 DOF motion of the UAV's camera will be applied to estimate the motion of the smartphones. By integrating the camera pose estimated by computer vision, and, by using the IMU sensors of the smartphones, the proposed method should more accurately measure the displacement of the object, even with the motion of the smartphone taken into consideration. The proposed method should enable the smartphone to be carried into a site and take video to measure displacements without connecting the camera to a tripod (Figure 63). The proposed work is also currently being conducted with Jaeho Shin.

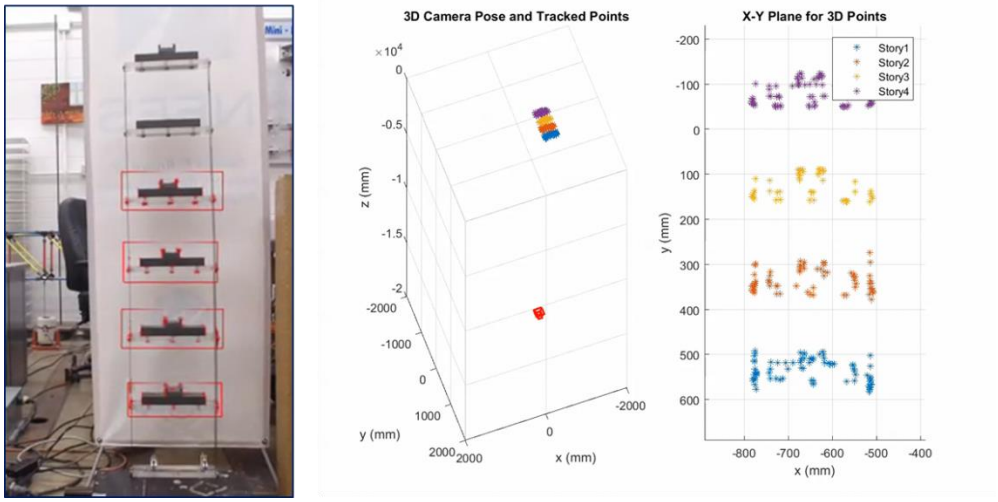


Figure 61: System Identification using UAVs.

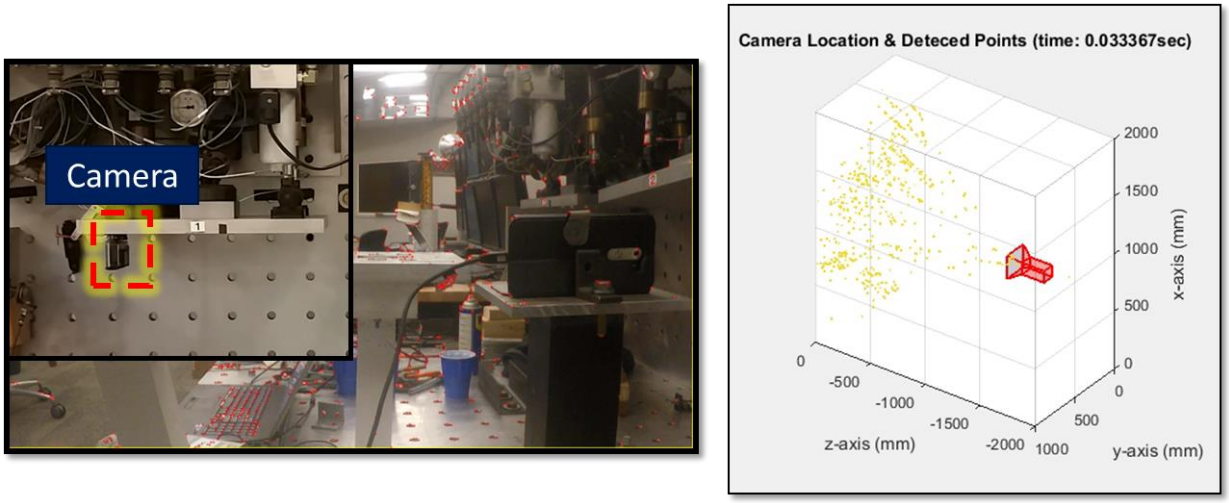


Figure 62: Structure Point-of-View Monitoring.

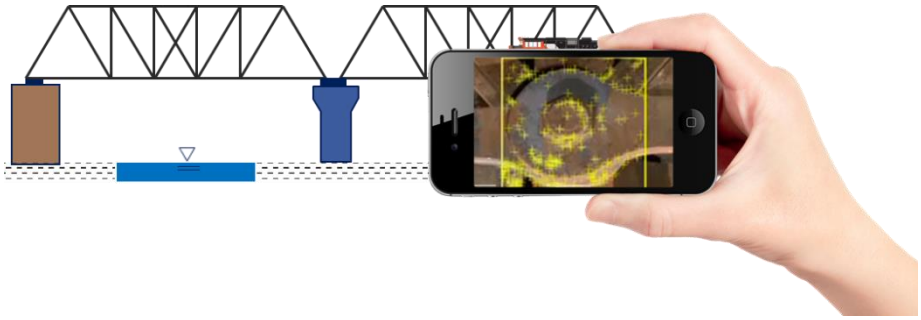


Figure 63: Enhanced Camera Pose Estimate using IMU Sensors.

Future of Smart Cities: Living Information Model

One of the directions for future work is to develop a "Living Information Model" (Figure 64). While the traditional building information model (BIM) is a static database often used for the design and construction periods, the proposed "Living Information Model" would be a model evolving and changing with the structure over its lifetime. The envisioned model will include the dynamic response (modal analysis of the structure and live load information) obtained from the information provided in chapters 5 and 6, and static damage of the structure (e.g., cracks and corrosion), together with the point cloud feature points constructed by UAVs. This 3D point cloud model will then associate the obtained information with the existing finite element models (FEM) and building information models (BIM), and in the end will visualize the real-time information of a structure's health as a layer upon these models.

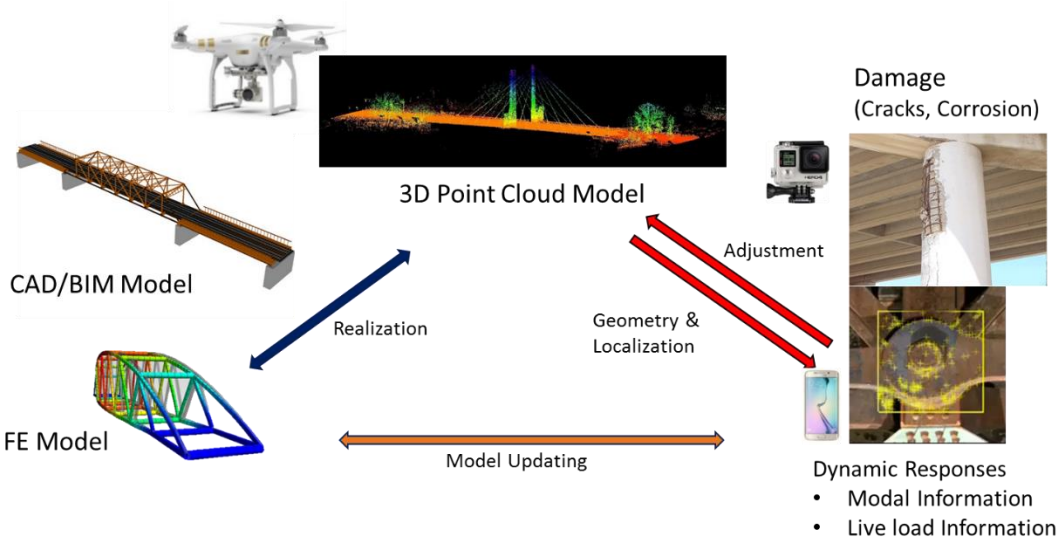


Figure 64: The Living Information Model.

REFERENCES

Akula, Manu, Suyang Dong, Vineet R Kamat, Lauro Ojeda, Adam Borrell, and Johann Borenstein. 2011. "Integration of infrastructure based positioning systems and inertial navigation for ubiquitous context-aware engineering applications." *Advanced Engineering Informatics* no. 25 (4):640-655.

Aoki, Tatsuo, Yoshihiro Fukumoto, Satoshi Yasuda, Yasuhiko Sakata, Kenta Ito, Jun Takahashi, Satoshi Miyata, Ichiro Tsuji, and Hiroaki Shimokawa. 2012. "The great East Japan earthquake disaster and cardiovascular diseases." *European heart journal*:ehs288.

ASCE. 2007. The Vision for Civil Engineering in 2025.

ASCE. 2013. Report Card for America's Infrastructure.

ATC. 2003. Users Manual: Mobile PostEarthquake

Building Safety Evaluation Data Acquisition

System. edited by Applied Technology Council. California.

Bahl, Paramvir, and Venkata N Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. Paper read at INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE.

Beauregard, Stephane., and Harald. Haas. 2006. Pedestrian Dead Reckoning: A Basis for Personal Positioning. Paper read at The 3rd Workshop on Positioning, Navigation and Communication.

bin Abdullah, Mohd Fikri Azli, Ali Fahmi Perwira Negara, Md Shohel Sayeed, Deok-Jai Choi, and Kalaiarasi Sonai Muthu. 2012. "Classification algorithms in human activity recognition using smartphones." *World Academy of Science, Engineering and Technology* no. 68:422-430.

Bose, Atreyi, and Chuan Heng Foh. 2007. A practical path loss model for indoor WiFi positioning enhancement. Paper read at Information, Communications & Signal Processing, 2007 6th International Conference on.

Bram, Jason, James Orr, and Carol Rapaport. 2002. "Measuring the effects of the September 11 attack on New York City." *Economic Policy Review* no. 8 (2).

Caetano, E, S Silva, and J Bateira. 2011. "A vision system for vibration monitoring of civil engineering structures." *Experimental Techniques* no. 35 (4):74-82.

Casper, Jennifer, and Robin R. Murphy. 2003. "Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center." *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* no. 33 (3):367-385.

Chang, CC, and XH Xiao. 2010. "An integrated visual-inertial technique for structural displacement and velocity measurement." *Smart Structures and Systems* no. 6 (9):1025-1039.

Chen, Xing Hua, and Piotr Omenzetter. 2013. "A framework for reliability assessment of an in-service bridge using structural health monitoring data." *Key Engineering Materials* no. 558:39-51.

Chengliang, Huang, Liao Zaiyi, and Zhao Lian. 2010. "Synergism of INS and PDR in Self-Contained Pedestrian Tracking With a Miniature Sensor Module." *Sensors Journal, IEEE* no. 10 (8):1349-1359. doi: 10.1109/jsen.2010.2044238.

Collin, J., O. Mezentsev, and G. Lachapelle. 2003. Indoor Positioning System Using Accelerometry and High Accuracy Heading Sensors. Paper read at The 16th International

Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003), at Portland, OR.

Collins, Steven H., and Arthur D. Kuo. 2013. "Two Independent Contributions to Step Variability during Over-Ground Human Walking." *PLoS ONE* no. 8 (8):e73597. doi: 10.1371/journal.pone.0073597.

Conly, G, and G Stewart. 1986. "Tragedy on the Track." *Grantham House, Wellington*.

Cover, Thomas, and Peter Hart. 1967. "Nearest neighbor pattern classification." *Information Theory, IEEE Transactions on* no. 13 (1):21-27.

Daniell, JE, B Khazai, and F Wenzel. 2013. "Uncovering the 2010 Haiti earthquake death toll." *Natural Hazards and Earth System Sciences Discussions* no. 1 (3):1913-1942.

Diebel, James. 2006. "Representing attitude: Euler angles, unit quaternions, and rotation vectors." *Matrix* no. 58:15-16.

Dirks, Susanne, Mary Keeling, and Jacob Dencik. 2009. "How smart is your city?: Helping cities measure progress." *IBM Institute for Business Value, IBM Global Business Services, New York*.

Duda, Richard O, Peter E Hart, and David G Stork. 1999. *Pattern classification*: John Wiley & Sons.

Evans, Dave. 2011. "The internet of things: How the next evolution of the internet is changing everything." *CISCO white paper* no. 1:1-11.

FEMA. 2004. Preliminar Damage Assessment Mnaual.

FEMA. 2006. Rescue Field Operations Guide.

FEMA. 2014. FEMA National US & R response system/structural collapse technician course: Appendix A: search capabilities.

Feng, Maria Q, Yoshio Fukuda, Dongming Feng, and Masato Mizuta. 2015. "Nontarget Vision Sensor for Remote Measurement of Bridge Dynamic Response." *Journal of Bridge Engineering*:04015023.

Fowler, Geoffrey A. 2015. Connecting Your Car to Your Smartphone Can Make Auto Data Work for You. *The Wall Street Journal*.

Fukuda, Yoshio, Maria Q Feng, Yuto Narita, Shun'ichi Kaneko, and Takayuki Tanaka. 2013. "Vision-based displacement sensor for monitoring dynamic response using robust object search algorithm." *Sensors Journal, IEEE* no. 13 (12):4725-4732.

Han, Jun., E. Owusu, L. T. Nguyen, A. Perrig, and J. Zhang. 2012. ACComplice: Location inference using accelerometers on smartphones. Paper read at Communication Systems and Networks (COMSNETS), 2012 Fourth International Conference on, 3-7 Jan. 2012.

Hatami, Ahmad. 2006. *Application of channel modeling for indoor localization using TOA and RSS*, Worcester Polytechnic Institute.

Hild, François, and Stéphane Roux. 2006. "Digital image correlation: from displacement measurement to identification of elastic properties–a review." *Strain* no. 42 (2):69-80.

HO, BL, and Rudolf E Kálmán. 1966. "Editorial: Effective construction of linear state-variable models from input/output functions." *at-Automatisierungstechnik* no. 14 (1-12):545-548.

Howarth, Fran. 2010. Security challenges in the US healthcare sector. *Bloor Research*.

James III, George H, Thomas G Carne, and James P Lauffer. 1993. "The natural excitation technique (NExT) for modal parameter extraction from operating wind turbines." *NASA STI/Recon Technical Report N* no. 93:28603.

Ji, YF, and CC Chang. 2008. "Nontarget image-based technique for small cable vibration measurement." *Journal of Bridge Engineering*.

Juang, J-N, and Richard S Pappa. 1985. "An eigensystem realization algorithm for modal parameter identification and model reduction." *Journal of guidance, control, and dynamics* no. 8 (5):620-627.

Kaemarungsi, Kamol, and Prashant Krishnamurthy. 2004. Properties of indoor received signal strength for WLAN location fingerprinting. Paper read at Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on.

Karsch, Kevin, Mani Golparvar-Fard, and David Forsyth. 2014. "ConstructAide: analyzing and visualizing construction sites through photographs and building models." *ACM Transactions on Graphics (TOG)* no. 33 (6):176.

Keally, Matthew, Gang Zhou, Guoliang Xing, Jianxin Wu, and Andrew Pyles. 2011. Pbn: towards practical activity recognition using smartphone-based body sensor networks. Paper read at Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems.

Khalil, Abdallah M. 2011. "Two-dimensional displacement measurement using static close range photogrammetry and a single fixed camera." *Alexandria Engineering Journal* no. 50 (3):219-227.

Khan, Adil Mehmood, Y-K Lee, SY Lee, and T-S Kim. 2010. Human activity recognition via an accelerometer-enabled-smartphone using kernel discriminant analysis. Paper read at Future Information Technology (FutureTech), 2010 5th International Conference on.

Kim, Sung-Wan, Bub-Gyu Jeon, Nam-Sik Kim, and Jong-Chil Park. 2013. "Vision-based monitoring system for evaluating cable tensile forces on a cable-stayed bridge." *Structural Health Monitoring* no. 12 (5-6):440-456.

King, AD. 1998. "Inertial navigation-forty years of evolution." *GEC review* no. 13 (3):140-149.

Knabb, Richard D, Jamie R Rhome, and Daniel P Brown. 2005. *Tropical cyclone report: Hurricane katrina, 23-30 august 2005*: National Hurricane Center.

Kohavi, Ron, and Foster Provost. 1998. "Glossary of terms." *Machine Learning* no. 30 (2-3):271-274.

Kothari, Nisarg, Balajee Kannan, Evan D Glasgwow, and M Bernardine Dias. 2012. "Robust indoor localization on a commercial smart phone." *Procedia Computer Science* no. 10:1114-1120.

Kuipers, Jack B. 1999. *Quaternions and rotation sequences*. Vol. 66: Princeton university press Princeton.

Ladd, Andrew M, Kostas E Bekris, Algis Rudys, Lydia E Kavraki, and Dan S Wallach. 2005. "Robotics-based location sensing using wireless ethernet." *Wireless networks* no. 11 (1-2):189-204.

Lee, Jong-Jae, Hoai-Nam Ho, and Jong-Han Lee. 2012. "A vision-based dynamic rotational angle measurement system for large civil structures." *Sensors* no. 12 (6):7326-7336.

Lee, Young-Seol, and Sung-Bae Cho. 2011. "Activity recognition using hierarchical hidden markov models on a smartphone with 3D accelerometer." In *Hybrid Artificial Intelligent Systems*, 460-467. Springer.

Li, Nan, Burcin Becerik-Gerber, Bhaskar Krishnamachari, and Lucio Soibelman. 2014. "A BIM centered indoor localization algorithm to support building fire emergency response operations." *Automation in Construction* no. 42:78-89.

Link, Jó Agila Bitsch, Paul Smith, Nicolai Viol, and Klaus Wehrle. 2011. Footpath: Accurate map-based indoor navigation using smartphones. Paper read at Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on.

Liu, Hongbo, Yu Gan, Jie Yang, Simon Sidhom, Yan Wang, Yingying Chen, and Fan Ye. 2012. Push the limit of wifi based localization for smartphones. Paper read at Proceedings of the 18th annual international conference on Mobile computing and networking.

Lucas, Bruce D, and Takeo Kanade. 1981. An iterative image registration technique with an application to stereo vision. Paper read at IJCAI.

Manyika, James, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela H Byers. 2011. "Big data: The next frontier for innovation, competition, and productivity."

Martin, Eladio, Oriol Vinyals, Gerald Friedland, and Ruzena Bajcsy. 2010. Precise indoor localization using smart phones. Paper read at Proceedings of the international conference on Multimedia.

Mok, Esmond, and Günther Retscher. 2007. "Location determination using WiFi fingerprinting versus WiFi trilateration." *Journal of Location Based Services* no. 1 (2):145-159.

Morlier, Joseph. 2011. "A pedagogical image processing tool to understand structural dynamics." In *Structural Dynamics, Volume 3*, 1215-1224. Springer.

Nogueira, F. M. A. , F. S. Barbosa, and L. P. S. Barra. 2005. "Evaluation of structural natural frequencies using image processing." *Proceedings of EVACES*.

Ojeda, L., and J. Borenstein. 2007. Personal Dead-reckoning System for GPS-denied Environments. Paper read at Safety, Security and Rescue Robotics, 2007. SSRR 2007. IEEE International Workshop on, 27-29 Sept. 2007.

Pai, D., M. Malpani, I. Sasi, N. Aggarwal, and P. S. Mantripragada. 2012. Padati: A Robust Pedestrian Dead Reckoning System on Smartphones. Paper read at Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on, 25-27 June 2012.

Parnandi, Avinash, Ken Le, Pradeep Vaghela, Aalaya Kolli, Karthik Dantu, Sameera Poduri, and Gaurav S Sukhatme. 2010. "Coarse in-building localization with smartphones." In *Mobile Computing, Applications, and Services*, 343-354. Springer.

Pei, Ling, Jingbin Liu, Robert Guinness, Yuwei Chen, Heidi Kuusniemi, and Ruizhi Chen. 2012. "Using LS-SVM based motion recognition for smartphone indoor wireless positioning." *Sensors* no. 12 (5):6155-6175.

Peña-Mora, Feniosky, ZUH Aziz, Albert Chen, Albert Plans, and Stuart Foltz. 2008. "Building assessment during disaster response and recovery." *Proceedings of the Institution of Civil Engineers* no. 161 (4):183-195.

Peña-Mora, Feniosky, Albert Y Chen, Zeeshan Aziz, Lucio Soibelman, Liang Y Liu, Khaled El-Rayes, Carlos A Arboleda, Timothy S Lantz Jr, Albert P Plans, and Sanyogita Lakhera. 2010. "Mobile ad hoc network-enabled collaboration framework supporting civil engineering emergency response operations." *Journal of Computing in Civil Engineering* no. 24 (3):302-312.

Pratama, A. R., Widyawan, and R. Hidayat. 2012. Smartphone-based Pedestrian Dead Reckoning as an indoor positioning system. Paper read at System Engineering and Technology (ICSET), 2012 International Conference on, 11-12 Sept. 2012.

Rantakokko, Jouni, Joakim Rydell, Peter Stromback, Peter Handel, Jonas Callmer, David Tornqvist, Fredrik Gustafsson, Magnus Jobs, and Mathias Gruden. 2011. "Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization." *Wireless Communications, IEEE* no. 18 (2):10-18.

Ravi, Nishkam, Nikhil Dandekar, Preetham Mysore, and Michael L Littman. 2005. Activity recognition from accelerometer data. Paper read at AAAI.

Riquier, Chris. 2015. "Reacting to the Connected Reality." *Research World* no. 2015 (54):36-38.

Rish, Irina. 2001. An empirical study of the naive Bayes classifier. Paper read at IJCAI 2001 workshop on empirical methods in artificial intelligence.

Schumacher, Thomas, and Ali Shariati. 2013. "Monitoring of Structures and Mechanical Systems Using Virtual Visual Sensors for Video Analysis: Fundamental Concept and Proof of Feasibility." *Sensors* no. 13 (12):16551-16564.

Shi, Jianbo, and Carlo Tomasi. 1994. Good features to track. Paper read at Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on.

Shih, Ming-Hsiang, and Wen-Pei Sung. 2013. "Developing dynamic digital image techniques with continuous parameters to detect structural damage." *The Scientific World Journal* no. 2013.

Spencer, Billie F, Manuel E Ruiz-Sandoval, and Narito Kurata. 2004. "Smart sensing technology: opportunities and challenges." *Structural Control and Health Monitoring* no. 11 (4):349-368.

Subbu, Kalyan Pathapati, Brandon Gozick, and Ram Dantu. 2013. "LocateMe: Magnetic-fields-based indoor localization using smartphones." *ACM Transactions on Intelligent Systems and Technology (TIST)* no. 4 (4):73.

Tian, Zengshan, Yuan Zhang, Mu Zhou, and Yu Liu. 2014. "Pedestrian dead reckoning for MARG navigation using a smartphone." *EURASIP Journal on Advances in Signal Processing* no. 2014 (1):1-9. doi: 10.1186/1687-6180-2014-65.

Tomasi, Carlo, and Takeo Kanade. 1991. *Detection and tracking of point features*: School of Computer Science, Carnegie Mellon Univ. Pittsburgh.

Torr, Philip HS, and Andrew Zisserman. 2000. "MLESAC: A new robust estimator with application to estimating image geometry." *Computer Vision and Image Understanding* no. 78 (1):138-156.

Tukey, John W. 1977. "Exploratory data analysis."

Van Haute, Tom, Eli De Poorter, Jen Rossey, Ingrid Moerman, Vlado Handziski, Arash Behboodi, Filip Lemic, Adam Wolisz, Niklas Wiström, and Thiemo Voigt. 2013. "The EVARILOS Benchmarking Handbook: Evaluation of RF-based Indoor Localization Solutions."

Webster, Peter J. 2008. "Myanmar's deadly daffodil." *Nature Geoscience* no. 1 (8):488-490.

Weiss, Gary M, and Jeffrey W Lockhart. 2012. The impact of personalization on smartphone-based activity recognition. Paper read at AAAI Workshop on Activity Context Representation: Techniques and Languages.

Yan, Zhixian, Vigneshwaran Subbaraju, Dipanjan Chakraborty, Archan Misra, and Karl Aberer. 2012. Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach. Paper read at Wearable Computers (ISWC), 2012 16th International Symposium on.

Yashinsky, Mark. 1998. The Loma Prieta, California, Earthquake of October 17, 1989-Highway Systems.

Yim, Jaegeol. 2013. "A smartphone indoor positioning method." *International Journal of Smart Home* no. 7 (5):9-18.

Zanca, Giovanni, Francesco Zorzi, Andrea Zanella, and Michele Zorzi. 2008. Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks. Paper read at Proceedings of the workshop on Real-world wireless sensor networks.

Zàruba, Gergely V, Manfred Huber, FA Kamangar, and Imrich Chlamtac. 2007. "Indoor location tracking using RSSI readings from a single Wi-Fi access point." *Wireless networks* no. 13 (2):221-235.

Zhang, Harry. 2004. "The optimality of naive Bayes." *AA* no. 1 (2):3.

Zhang, Z. Y. 2000. "A flexible new technique for camera calibration." *Ieee Transactions on Pattern Analysis and Machine Intelligence* no. 22 (11):1330-1334. doi: Doi 10.1109/34.888718.